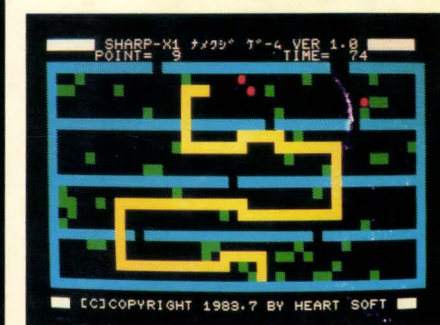
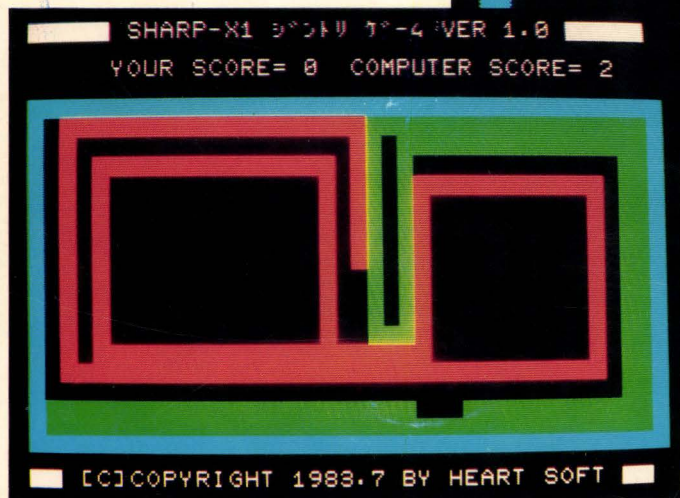
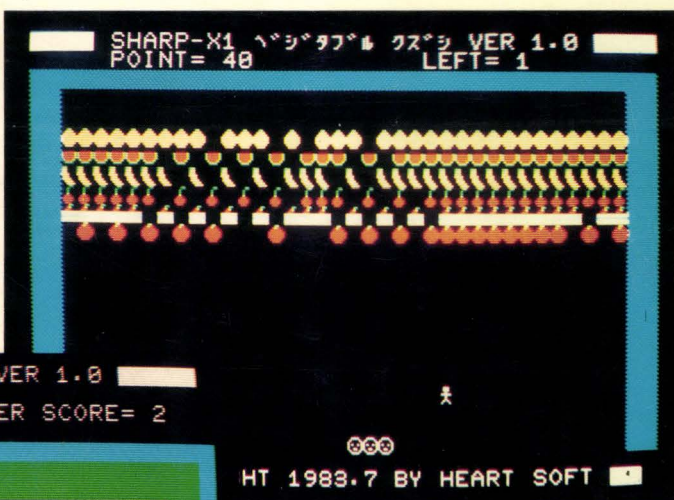
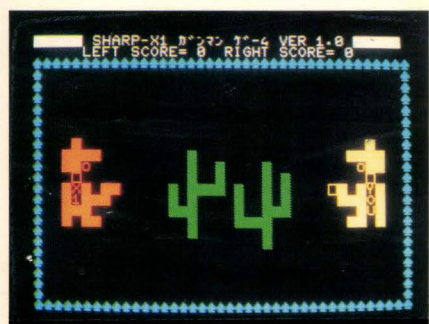
BASIC　ソースブック

パソコンテレビX1ゲームプログラム(II)

ハートソフト編

ギャラクシーウォーズ／ガンマンゲーム／キャラクターメーカー／デフレクション／日本語エラーメッセージ／インベーダーたたき／TVテニス／陣取りゲーム／砲撃ゲーム／算数教室／オートデータメーカー／BLACK　JACK／自動簡易翻訳機／ベジタブルくずし／ナメクジゲーム／コンピュータ家計簿／競馬ゲーム／ハングマン／バイオリズム／メロディーメーカー

工学図書株式会社刊

BASIC ソースブックス

パソコンテレビX1ゲームプログラム(II)

工学図書株式会社

前書き

Ｘ１でプログラムを作ることになって，気が付いたことですが，第一印象としては，『Ｘ１って随分贅沢な機械だ』ということです。

一度に，あの機械，この機械と複数のマイコンを使っていると，〃プログラムを作る，走らせる，すると，ピーと音が！　Syntax error が出る，あれ？　と思って，LIST を取る。すると別におかしい所が見当らない……。？？ あァ！　そうか，この命令は，この機械には無かったんだ！.！〃 ということが応々にしてありますが．Ｘ１を使っている間は一度も，このような状態に遭遇することは，ありませんでした。Ｘ１は，国産の Hu-BASIC を使っていますが，これは，今ほとんどのパソコンに塔載されている，米国マイクロソフト社の BASIC の良い所や，またそれ以上に良い命令を持っています。」これほど多彩なBASICは，ないのではないか」と言っても過言ではないと思います。 HARD にしても，64Ø×2ØØ ドットのフルカラー， キャラクターゼネレータ機能，AY-3-819Øを使ったミュージック機能など，いたりつくせりです。
また，パソコンテレビと言われるように，今までのパソコンには考えられなかった，ＴＶとコンピュータ画面の合成，ＴＶチャンネル，音量のコントロール，ＴＶ番組予約などの機能も盛りだくさんです。ただ心配なのは，「これだけの機能のパソコンを，使いこなせるだろうか？」ということです。これだけのパソコンをＴＶを見ているだけでは，もったいない！！，早く自分の思いのままに動かせるようになって下さい。

思いのままに操作できるようになるためには，どうしたら良いか？，というと，まずＸ１に触りまくることです。プログラムを入力するだけでもかまいませんが，「この命令は，どういうことをしているのだろう」という疑問から，１歩１歩理解していくものだと思います。

本書は，マイコンをちょっとかじった人，初心者でも気楽に読めるように，各プログラムには，フローチャート(流れ図)，変数表，リストには各ルーチンの説明を加えました。また，プログラムはすべて，手元にあるマイコンにかけてテストしてあります。どうぞ，御参考にして下さい。

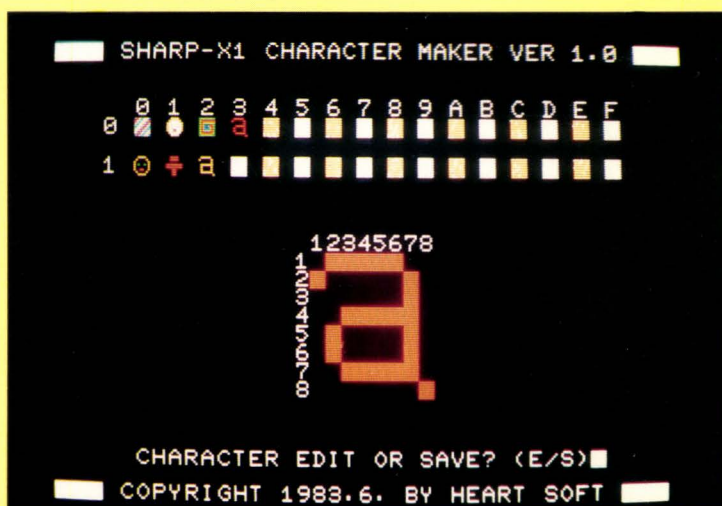
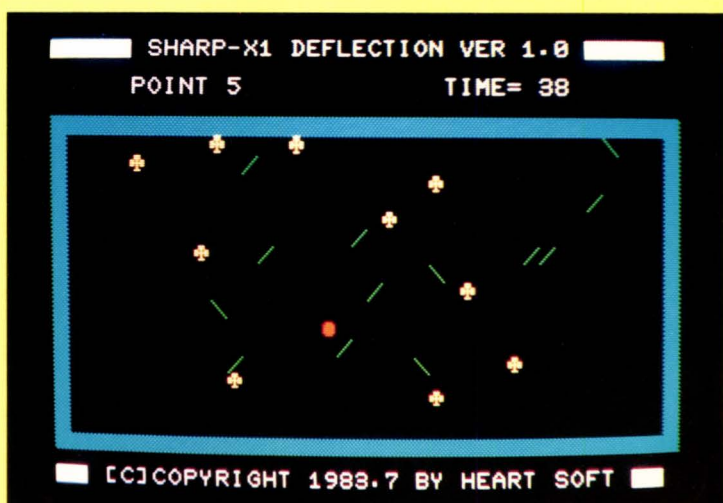
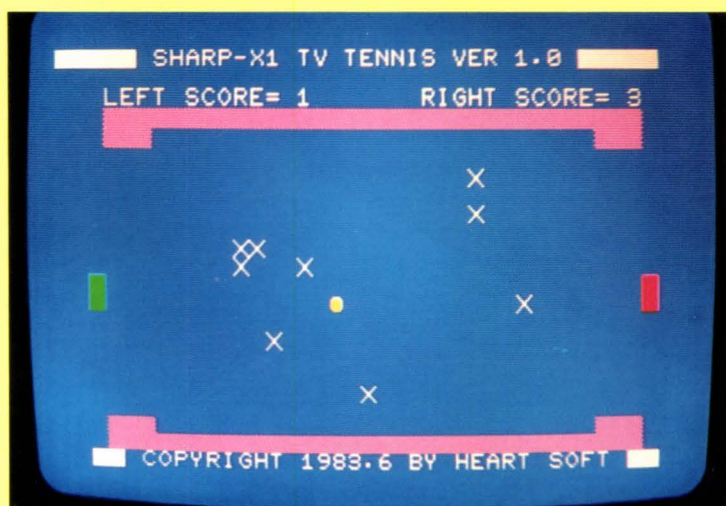
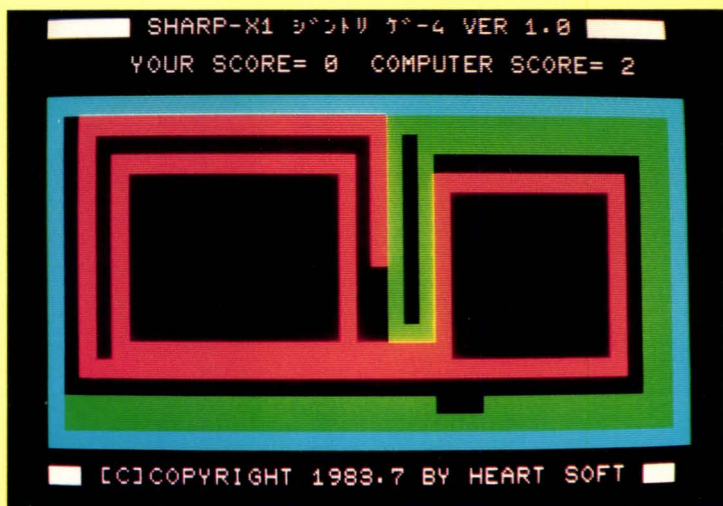
なお，出版にあたっては，工学図書の方々に大変お世話になり，とくに同社，笠原進氏には，多大な御協力を得ました，記して感謝の意を表します。

HART　SOFT

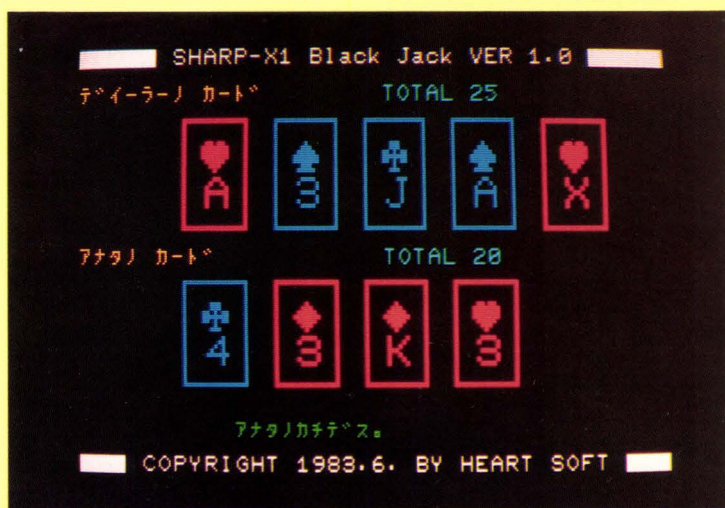
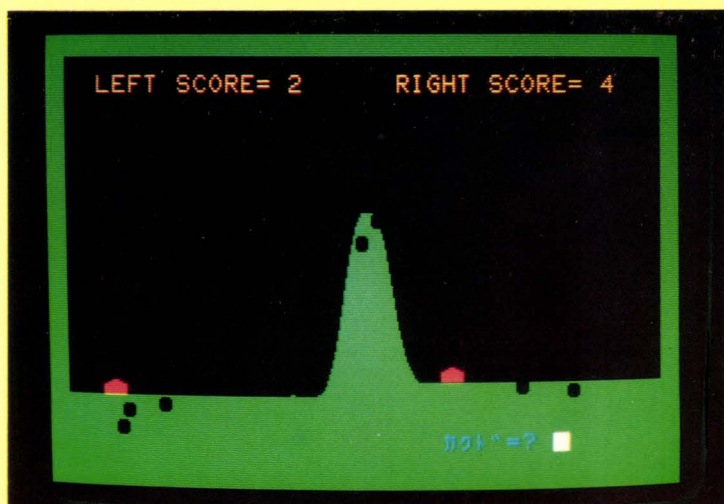
目　　次

ギャラクシーウォーズ
（1ページ）

ガンマンゲーム
（6ページ）

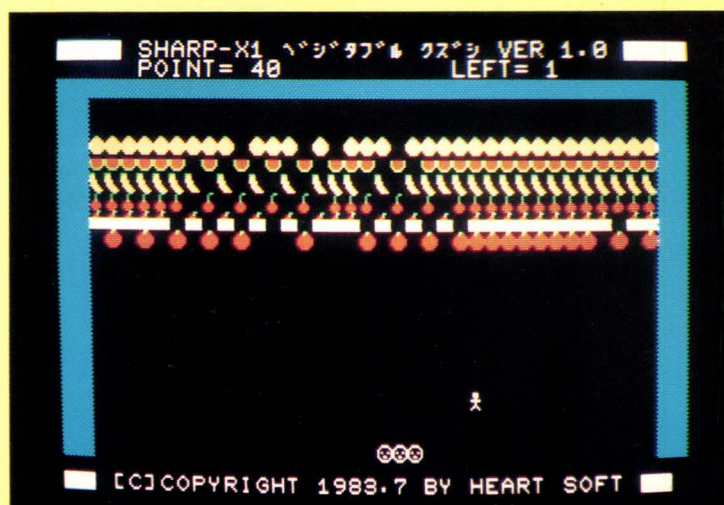
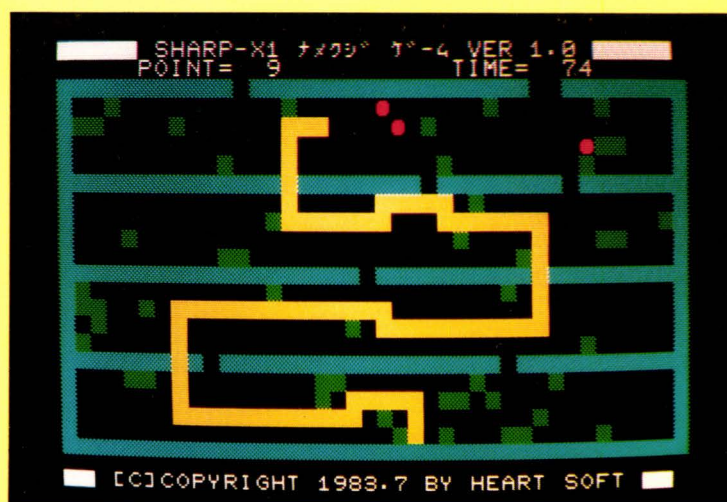
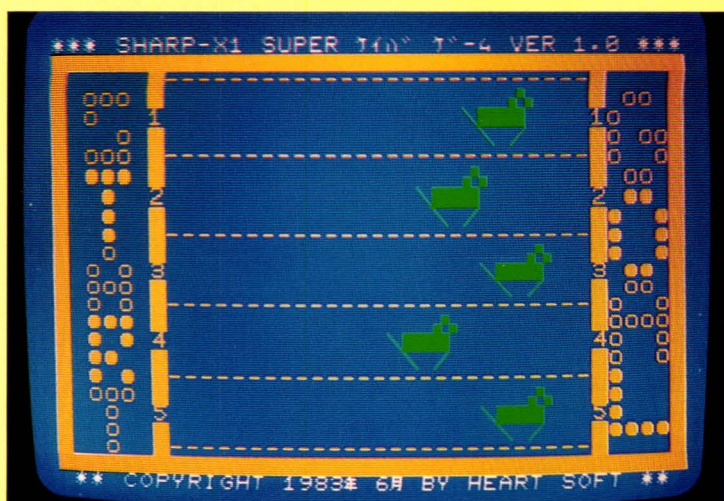
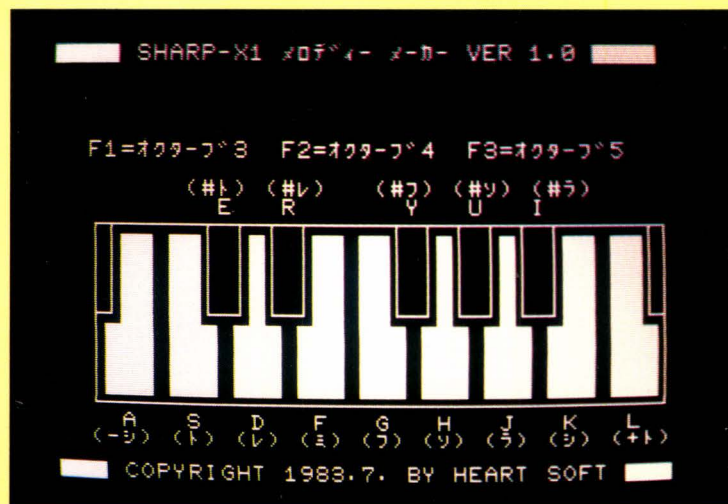
キャラクターメーカー
（12ページ）

デフレクション
（20ページ）

ＴＶテニス
（34ページ）

陣取りゲーム
（39ページ）

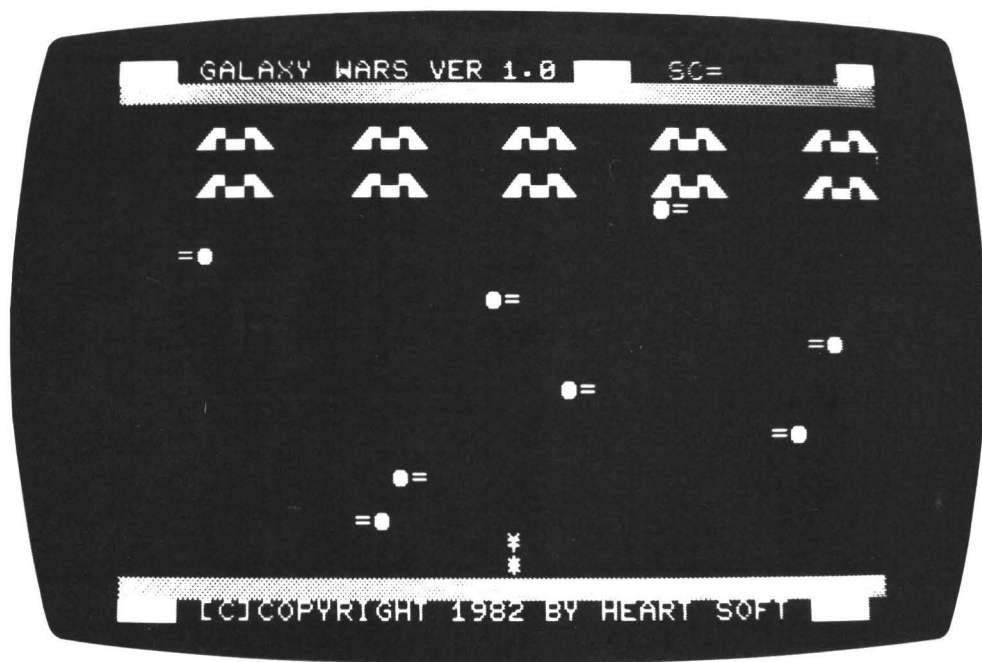
砲撃ゲーム

（44ページ）

BLACK JACK

（57ページ）

ベジタブルくずし

（67ページ）

ナメクジゲーム
（72ページ）

競馬ゲーム
（89ページ）

メロディーメーカー
（103ページ）

1
ギャラクシーウォーズ

一時期ものすごいインベーダーゲームのブームがありました。町を歩けばあちらこちらからインベーダー特有の電子音が聞こえて来たものです。その頃大量に生産されたインベーダー機はブームがさってからどうなったか？ というと，ROM を変更されて GALAXY WARSやルナーレスキュー，ルパン3世などに変身したようです。そこで，あまり日の目を見なかった GALAXY WARS を少々スピードは遅いですが，作ってみました。

ゲームの遊び方

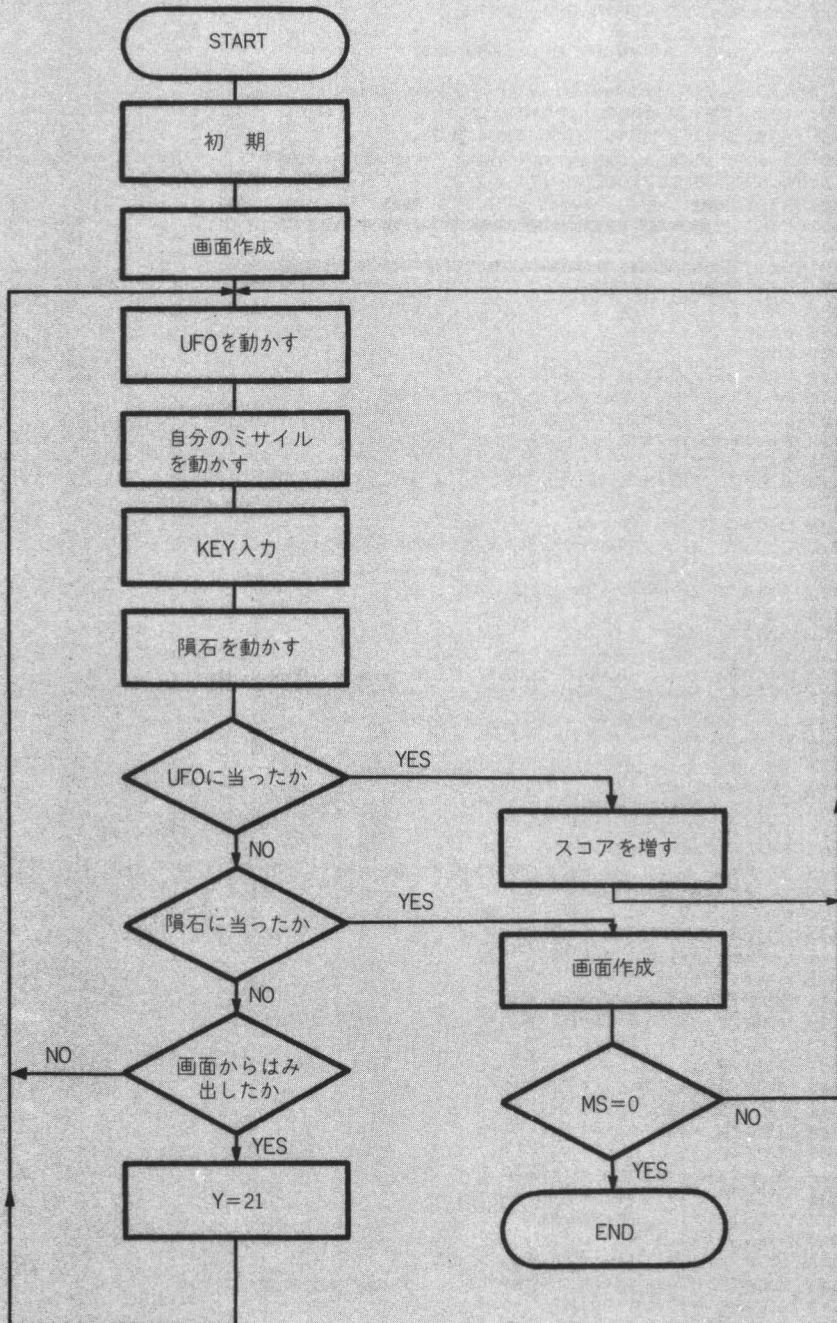
このゲームは，途中の隕石に当たらないように，ミサイルを操作して敵のＵＦＯに体当りするものです。本物とのちがいは，ＵＦＯが放射能を出してきません。これをやらないのは，極端にスピードが落ちてしまうためです。操作キーは，発射＝テンキー８　左移動＝テンキー４　右移動＝テンキー６　となっています。ＵＦＯにうまく当たれば 100～1000 までの点数が加算されます。なお隕石に当ったかの判定は“￥”のみで行っており，炎“＊”では判定は行いません。ミサイルが隕石に５回当たるとゲームオーバーです。

高得点の取り方

発射キーを押すまでスタートしませんから充分相手ＵＦＯの動きを読んでから発射しましょう。なおＵＦＯのどこに当っても点数は同じです。端っこでも当っておきましょう。

変数表

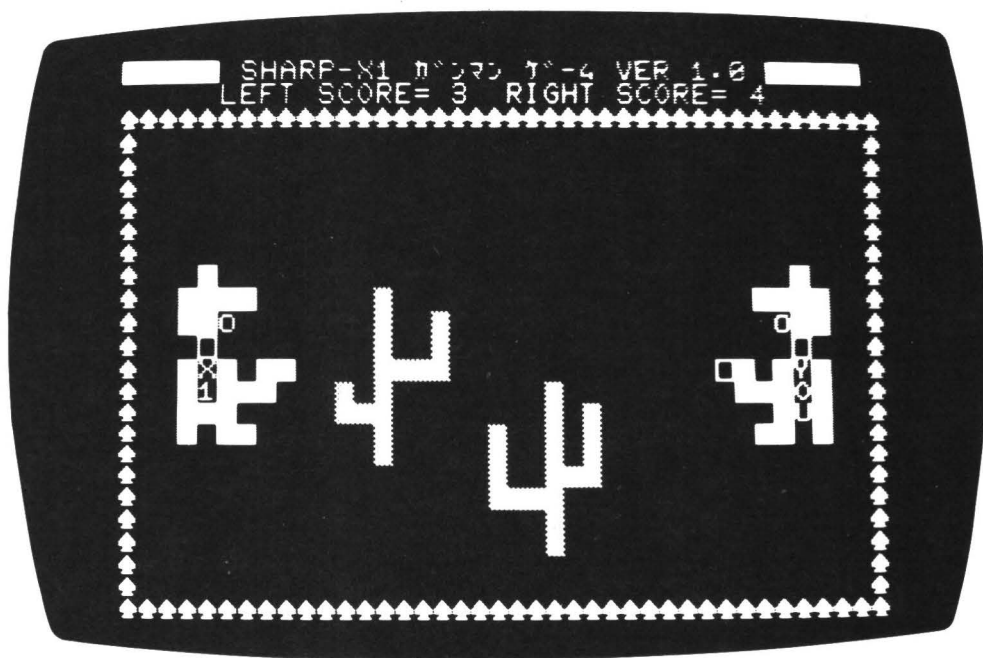
Ｘ，Ｙ→ミサイルのＸ，Ｙ座標
ＸＸ，ＹＹ→移動用仮変数
ＳＣ→スコア
ＭＳ→ミサイルの数
Ｐ(ｎ)→隕石の位置
Ｅ＄(０)，Ｅ＄(１)→ＵＦＯのデータ
Ｋ→画面読み取り用
Ｉ＄→キー入力用

ギャラクシーウォーズ・フローチャート
START
初　期
画面作成
UFOを動かす
自分のミサイルを動かす
KEY入力
隕石を動かす
UFOに当ったか
YES
NO
スコアを増す
隕石に当ったか
YES
NO
画面作成
画面からはみ出したか
NO
YES
Y=21
MS=0
NO
YES
END

1. ギャラクシーウォーズ

```
1 REM*************************************
2 REM*                                   *
3 REM*  ■■■ GALAXY WARS VER 1.0 ■■■      *
4 REM*                                   *
5 REM*      [C] COPYRIGHT 1983年 7月     *
6 REM*                                   *
7 REM*   FOR SHARP-X1 BY HEART SOFT      *
8 REM*                                   *
9 REM*************************************
10 WIDTH 40:CONSOLE 0,25:CLICK OFF
20 X=20:Y=21:XX=X:YY=Y:S=0:MS=5:SC=0                     ─UFOのデータ
30 E$(0)="▟▙    ▟▙    ▟▙    ▟▙    ▟▙      ":E$(1)=E$(0)
40 COLOR 2,5:CLS:COLOR 1
50 PRINT"■■■ GALAXY WARS VER 1.0 ■■■  SC=      ■■";     ┐
60 PRINT"▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓";      │
70 LOCATE 0,23                                          │ 画面作成
80 PRINT"▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓";      │
90 PRINT"■■■ [C]COPYRIGHT 1982 BY HEART SOFT ■■■";      ┘
100 FOR I=1 TO 8:P(I)=INT(RND(1)*40):NEXT I
110 REM                                      ─●この位置を決める
120 COLOR 6
130 LOCATE 0,3:PRINTE$(0)                               ┐
140 E$(0)=RIGHT$(E$(0),1)+LEFT$(E$(0),39)               │ UFOを動かすルーチン
150 LOCATE 0,5:PRINT E$(1)                              │
160 E$(1)=RIGHT$(E$(1),39)+LEFT$(E$(1),1)               ┘
170 REM
180 MUSIC "O6":SOUND 7,&HF8
190 COLOR 2                                             ┐
200 LOCATE XX,YY:PRINT" ";                              │
210 LOCATE X,Y:PRINT"¥";CHR$(31);CHR$(29);"*";          │ 自分のミサイルを
                      ↖本体                 ↖炎          │  動かすルーチン
220 PAUSE 2                                             │
230 LOCATE X,Y+1:PRINT" ";                              ┘
240 REM
250 II=STICK(0)                                         ┐
260 IF II=8 THEN S=1                                    │            8→ 発射
270 IF S=0 THEN  MUSIC "A0R4" ELSE GOSUB 1290           │ キー入力   4→ 左移動
280 IF II=4 THEN XX=X:YY=Y:X=X-1:GOTO 310               │            6→ 右移動
290 IF II=6 THEN XX=X:YY=Y:X=X+1:GOTO 310               ┘
300 XX=X:YY=Y
310 Y=Y-S
320 IF Y<3 THEN Y=21
330 IF X<=3 THEN X=3
340 IF X>=36 THEN X=36
350 K$=CHARACTER$(X,Y)
360 IF K$="◢" OR K$="◣" OR K$="▞" OR K$="▚" THEN 730 ── UFOに当ったとき
370 IF K$="●" OR K$="=" THEN 940    ──隕石に当ったとき
380 REM
390 COLOR 2                                             ┐
400 LOCATE P(1),6:PRINT"  ";                            │
410 P(1)=P(1)-1                                         │
420 IF P(1)<=1 THEN P(1)=38                             │
430 LOCATE P(1),6:PRINT"●=";                            │
440 LOCATE P(2),10:PRINT"  ";                           │
450 P(2)=P(2)-1                                         │
460 IF P(2)<=1 THEN P(2)=38                             │
470 LOCATE P(2),10:PRINT"●=";                           │
480 LOCATE P(3),14:PRINT"  ";                           │ 左へ寄り切ったら右へ戻す
490 P(3)=P(3)-1                                         │
500 IF P(3)<=1 THEN P(3)=38                             │
510 LOCATE P(3),14:PRINT"●=";                           │
520 LOCATE P(4),18:PRINT"  ";                           │
530 P(4)=P(4)-1                                         │
540 IF P(4)<=1 THEN P(4)=38                             │
550 LOCATE P(4),18:PRINT"●=";                           │ 隕石を動かすルーチン
560 LOCATE P(5),8:PRINT"  ";                            │
570 P(5)=P(5)+1                                         │
580 IF P(5)>=38 THEN P(5)=1                             │
590 LOCATE P(5),8:PRINT"=●";                            │
600 LOCATE P(6),12:PRINT"  ";                           │
610 P(6)=P(6)+1                                         │
620 IF P(6)>=38 THEN P(6)=1                             │ 右へ寄り切ったら左へ戻す
630 LOCATE P(6),12:PRINT"=●";                           │
640 LOCATE P(7),16:PRINT"  ";                           │
650 P(7)=P(7)+1                                         │
660 IF P(7)>=38 THEN P(7)=1                             ┘
```

```
670 LOCATE P(7),16:PRINT"=●";
680 LOCATE P(8),20:PRINT"   ";
690 P(8)=P(8)+1
700 IF P(8)>=38 THEN P(8)=1
710 LOCATE P(8),20:PRINT"=●";
720 GOTO 110
730 REM ｴﾝﾊﾞﾝ ﾆ ｱﾀﾙ
740 COLOR 1
750 GOSUB 1340:W=W+1:IF W=10 THEN 1170
760 R=INT(RND(1)*10)*100+100
770 LOCATE X-2,Y
780 PRINT USING"####",R
790 SC=SC+R
800 LOCATE 32,0
810 PRINT USING"#####",SC
820 PAUSE 10
830 LOCATE X-3,Y:PRINT"          ";
840 IF Y=3 THEN 860
850 IF Y=5 THEN 900
860 E$(0)="":FOR I=0 TO 39
870 E$(0)=E$(0)+CHARACTER$(I,3)
880 NEXT I
890 Y=21:S=0:GOTO 110
900 E$(1)="":FOR I=0 TO 39
910 E$(1)=E$(1)+CHARACTER$(I,5)
920 NEXT I
930 Y=21:S=0:GOTO 110
940 REM ｲﾝｾｷ ﾆ ｱﾀﾙ
950 GOSUB1380:BEEP:COLOR 2
960 LOCATE X-1,Y-1:PRINT"***";
970 LOCATE X-1,Y  :PRINT"***";
980 LOCATE X-1,Y+1:PRINT"***";
990 PAUSE 20
1000 LOCATE X-1,Y-1:PRINT"   ";
1010 LOCATE X-1,Y  :PRINT"   ";
1020 LOCATE X-1,Y+1:PRINT"   ";
1030 X=20:Y=21:XX=X:YY=Y
1040 COLOR 6
1050 MS=MS-1:IF MS<>0 THEN 110
1060 REM
1070 COLOR 6
1080 LOCATE 0,12
1090 PRINT
1100 PRINT"               ｹﾞｰﾑ ｵｰﾊﾞｰ
1110 PRINT
1120 PRINT"            ﾓｳ1ﾄﾞ ｱｿﾋﾞﾏｽｶ (Y/N)?";
1130 I$=INPUT$(1)
1140 IF I$="Y" OR I$="y" THEN RUN
1150 IF I$="N" OR I$="n" THEN CLS:END
1160 GOTO 1130
1170 REM
1180 COLOR 6
1190 LOCATE 0,12
1200 BO=INT(RND(1)*10)*100+100
1210 SC=SC+BO
1220 PRINT
1230 PRINT"               ｵｰﾙ ｸﾘｱｰ
1240 PRINT
1250 PRINT"      ﾂｷﾞﾓ ｶﾞﾝﾊﾞﾚ! ﾎﾞｰﾅｽ ";:PRINT USING"####";BO;
1260 PAUSE 40
1270 X=20:Y=21:XX=X:YY=Y:W=0
1280 GOTO 30
1290 REM ﾛｹｯﾄ ﾉ ｵﾄ
1300 SOUND 0,200:SOUND 1,31:SOUND 2,100:SOUND 3,4
1310 SOUND 4,100:SOUND 5,30:SOUND 6,2 :SOUND 7,&HC7
1320 SOUND 8,12:SOUND 9,12:SOUND 10,12:SOUND 11,100
1330 SOUND 12,10:SOUND 13,13:RETURN
1340 REM ｴﾝﾊﾞﾝ ﾆ ｱﾀｯｸ ﾄｷ ﾉ ｵﾄ
1350 SOUND 0,200:SOUND 1,0:SOUND 2,100:SOUND 3,1:SOUND 4,200
1360 SOUND 5,20:SOUND 6,31:SOUND 7,&HFC:SOUND 8,16:SOUND 9,16
1370 SOUND 10,16:SOUND 11,200:SOUND 12,10:SOUND 13,12:RETURN
1380 REM ﾛｹｯﾄ ﾆ ｱﾀｯｸ ﾄｷ ﾉ ｵﾄ
1390 SOUND 0,200:SOUND 1,0:SOUND 2,100:SOUND 3,1:SOUND 4,200
1400 SOUND 5,20:SOUND 6,31:SOUND 7,&HE5:SOUND 8,16:SOUND 9,16
1410 SOUND 10,16:SOUND 11,200:SOUND 12,20:SOUND 13,0 :RETURN
```

- 730–930: UFOに当ったルーチン
  - 760: 得点をランダムに決める
  - 770–810: 数値出力
  - 820: 時間待ち
  - 860–930: 新たにUFOのデータを読み込む
- 950–1050: 隕石に当ってしまったとき
- 1070–1150: ゲームオーバー
  - 1140–1150: 再ゲームか？　Y → RUN　N → END
- 1200: ボーナス点をランダムに決定する
- 1210–1270: 一面クリアー
- 1290–: 以下音のルーチン

2
ガンマンゲーム

久しぶりに西部劇の深夜映画を見たのですが，やはりカッコイイものです。悪人相手に銃で戦う所なんかは最高です。気分はもう，ガンマン！

そこで，Ｘ１でもと思い作ったのが，このガンマンゲームです。保安官になったつもりで遊べば，また変わった気分になれるかもしれません。さあ，悪人相手に決闘です！

ゲームの遊び方

あなたのガンマンは右側です。操作キーはテンキーの2と8で上下に動きスペースキーで発射です。途中に障害物として，サボテンが2本あり，そこに弾が当っても弾はストップしてしまい相手に届きません。相手は自分が1歩進むとき 0.5歩しか進めませんが，急所が1ヶ所しかありません。逆に自分は敵より速く動けますが急所が2ヶ所あります。急所に弾が当たりますと首が飛び，どちらかが，5回撃たれるとゲームオーバーです。

ガンマン変数表

GM$(0)～GM$(9)→敵のガンマン

GN$(0)～GN$(9)→自分のガンマン

SA$(0)～SA$(9)
SB$(0)～SB$(9)　}サボテンのデータ

X1，Y1→敵のガンマンのX，Y座標

X2，Y2→自分の　〃　〃　〃

X，Y→サボテンのX，Y座標

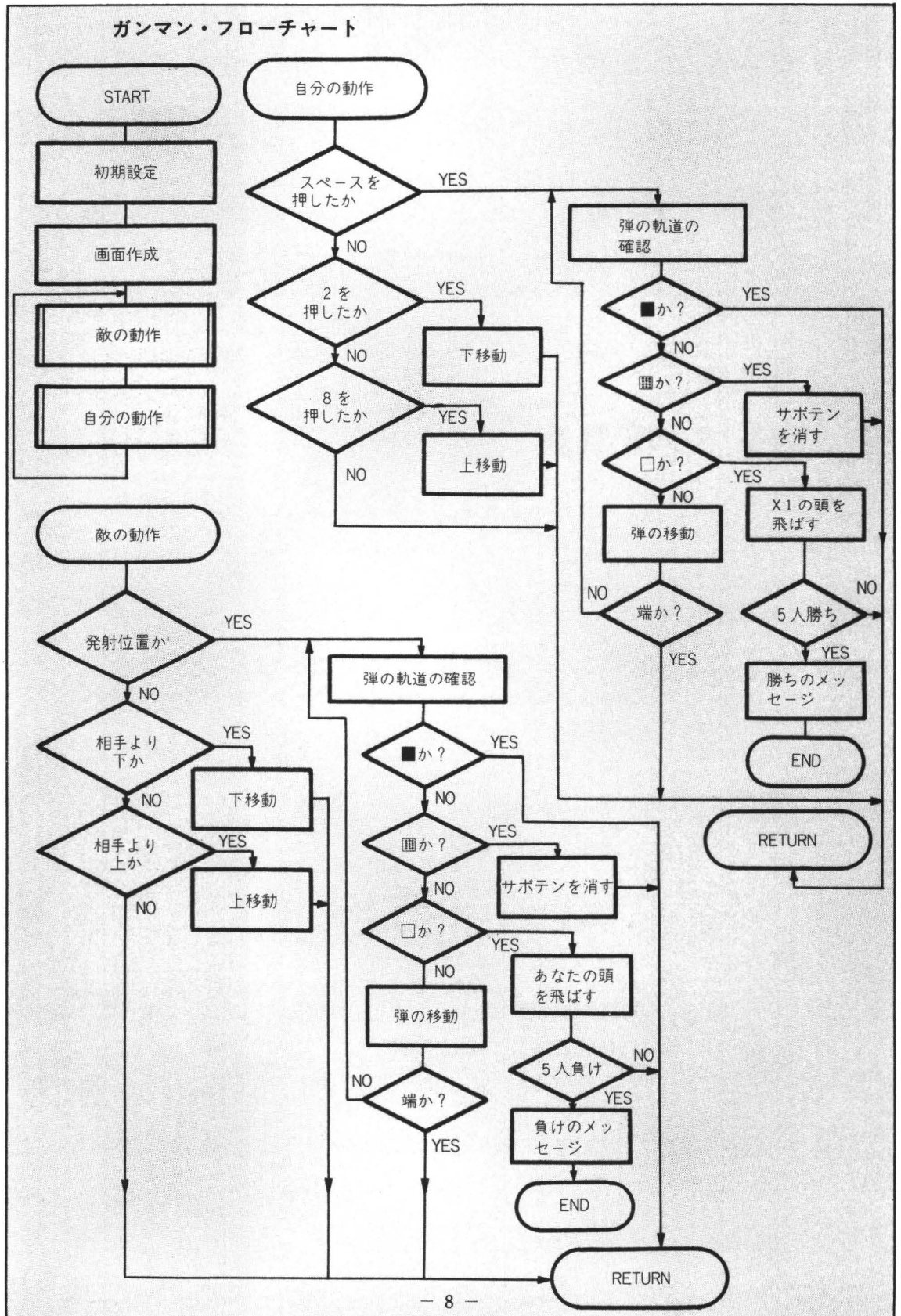
RS→自分の得点

LS→敵の得点

P$→画面読み取り

I$→キー入力用

I→弾のX座標，ループ用に使用

2．ガンマンゲーム

ガンマン・フローチャート
START
初期設定
画面作成
敵の動作
自分の動作
自分の動作
スペースを押したか
YES
NO
2を押したか
YES
下移動
NO
8を押したか
YES
上移動
NO
弾の軌道の確認
■か？
YES
NO
▦か？
YES
サボテンを消す
NO
□か？
YES
X1の頭を飛ばす
NO
弾の移動
端か？
NO
YES
5人勝ち
NO
YES
勝ちのメッセージ
END
RETURN
敵の動作
発射位置か
YES
NO
相手より下か
YES
下移動
NO
相手より上か
YES
上移動
NO
弾の軌道の確認
■か？
YES
NO
▦か？
YES
サボテンを消す
NO
□か？
YES
あなたの頭を飛ばす
NO
弾の移動
端か？
NO
YES
5人負け
NO
YES
負けのメッセージ
END
RETURN

```
1 REM*************************************
2 REM*                                   *
3 REM*  ■■■■ ｶﾞﾝﾏﾝ ｹﾞｰﾑ VER 1.0 ■■■■    *
4 REM*                                   *
5 REM*      [C]Copyright 1983年 7月      *
6 REM*                                   *
7 REM*   FOR SHARP-X1 By Heart Soft      *
8 REM*                                   *
9 REM*************************************
10 CONSOLE 0,25:WIDTH 40:CLICK OFF
20 RANDOMIZE:COLOR 7:TEMPO 700
30 GM$(0)="          ":GN$(0)="          "
40 GM$(1)="  ■       ":GN$(1)="       ■  "
50 GM$(2)=" ■■■■     ":GN$(2)="     ■■■■ "
60 GM$(3)=" ■o       ":GN$(3)="       o■ "
70 GM$(4)="  □       ":GN$(4)="       □  "
80 GM$(5)=" ■X■ ■■   ":GN$(5)="□■ ■Y■    "
90 GM$(6)=" ■1■■     ":GN$(6)="     ■O■  "
100 GM$(7)="  ■■■     ":GN$(7)="     ■U■  "
110 GM$(8)="  ■ ■■    ":GN$(8)="    ■■ ■  "
120 GM$(9)="          ":GN$(9)="          "
```

ガンマンのデータ
自由に変えても良いが，□印のところだけは変えないように

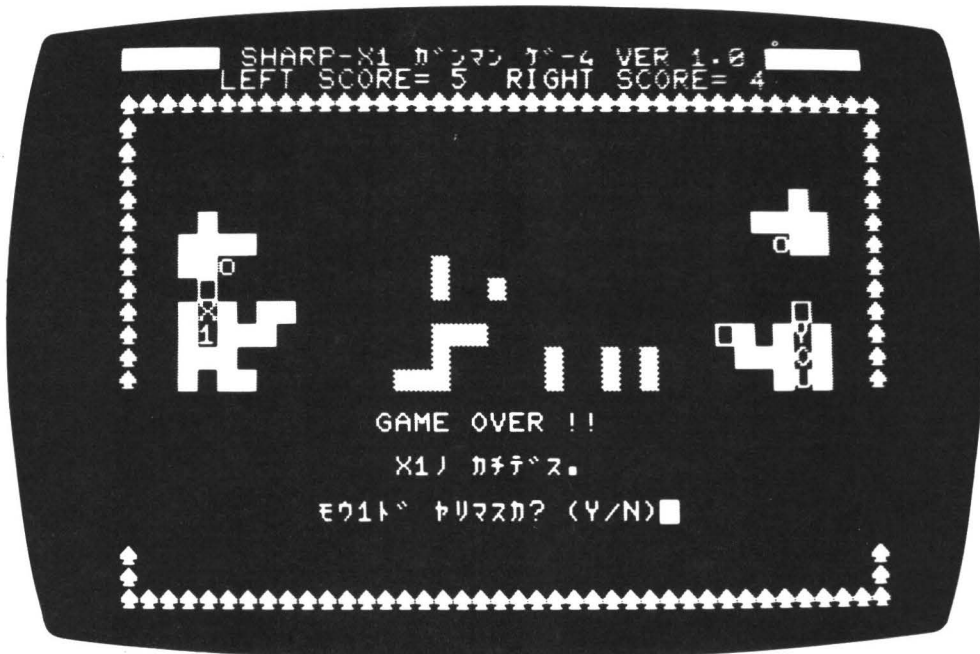
```
130 REM
140 SA$(0)="          ":SB$(0)="          "
150 SA$(1)="   ■      ":SB$(1)="      ■   "
160 SA$(2)="   ■  ■   ":SB$(2)="     ■ ■  "
170 SA$(3)="   ■  ■   ":SB$(3)="  ■  ■ ■  "
180 SA$(4)=" ■ ■■■■   ":SB$(4)="  ■  ■ ■  "
190 SA$(5)=" ■ ■      ":SB$(5)="  ■  ■■■  "
200 SA$(6)=" ■■■■     ":SB$(6)="  ■■■■    "
210 SA$(7)="   ■      ":SB$(7)="     ■    "
220 SA$(8)="   ■      ":SB$(8)="     ■    "
230 SA$(9)="          ":SB$(9)="          "
```

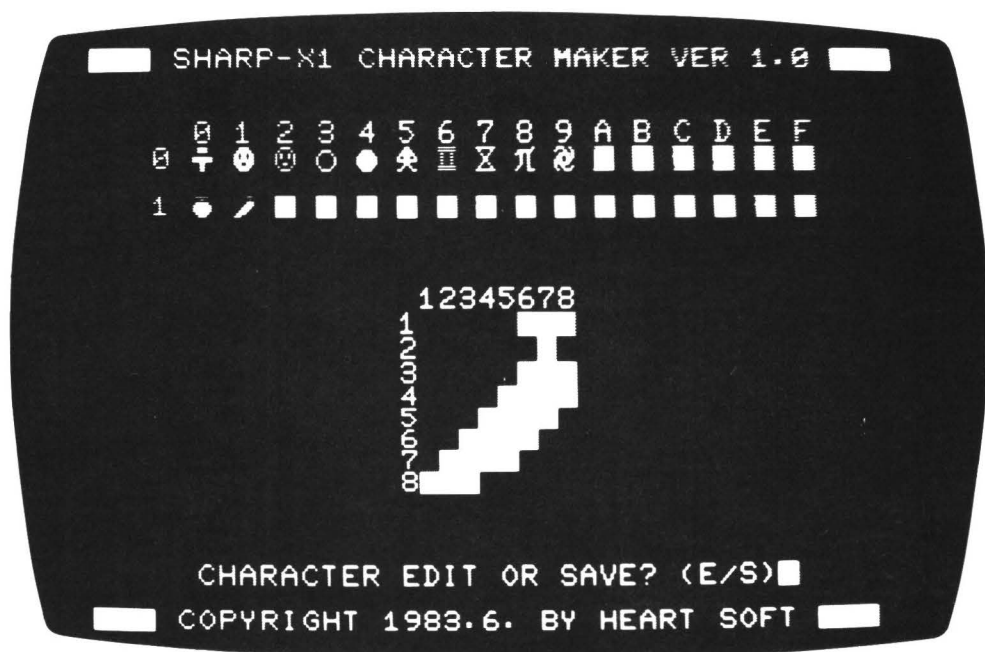
サボテンのデータ
自由に変えても良いが，同じキャラクターを使用すること

```
240 CLS:COLOR7:PRINT"■■■■ SHARP-X1 ｶﾞﾝﾏﾝ ｹﾞｰﾑ VER 1.0 ■■■■";
250 LOCATE 0,24:PRINT"■■■ COPYRIGHT 1983.7. BY HEART SOFT ■■■";
260 LOCATE  5,1:PRINT"LEFT SCORE=";LS
270 LOCATE 20,1:PRINT"RIGHT SCORE=";RS
280 COLOR 5
290 LINE(0,2)-(39,24),"♠",B         ――― 枠を出力する
300 COLOR 4
310 X=RND(1)*5+8 :Y=RND(1)*5+4      ┐
320 FOR I=0 TO 9                    │
330 LOCATE X,Y+I                    │ 左側のサボテンを出力
340 PRINT SA$(I);                   │
350 NEXT I                          ┘
360 X=RND(1)*5+18:Y=RND(1)*5+9      ┐
370 FOR I=0 TO 9                    │
380 LOCATE X,Y+I                    │ 右側のサボテンを出力
390 PRINT SB$(I);                   │
400 NEXT I                          ┘
410 X1=2:Y1=8:X2=30:Y2=8
420 GOSUB 570                       ――― 敵のガンマンを出力
430 GOSUB 640                       ――― 自分のガンマンを出力
440 REM
450 IF STRIG(0)=-1 THEN 710         ――― スペースを押したとき
460 IF STICK(0)=2 THEN Y2=Y2+1      ┐ キー入力　　2で下がり8で上がる
470 IF STICK(0)=8 THEN Y2=Y2-1      ┘
480 IF Y2<3 THEN Y2=3               ┐ はみ出し処理
490 IF Y2>14 THEN Y2=14             ┘
500 GOSUB 640
510 REM
520 IF Y1-Y2=0 OR Y1-Y2=-1 THEN 890 ――― 発射ルーチンへ
530 IF Y1>Y2-1 THEN Y1=Y1-.5        ┐ 敵のガンマンの移動ルーチン
540 IF Y1<Y2-1 THEN Y1=Y1+.5        ┘
550 GOSUB 570
560 GOTO 440
570 REM
580 COLOR 2                         ┐
590 FOR I=0 TO 9                    │
600 LOCATE X1,Y1+I                  │ 敵のガンマンを出力するルーチン
610 PRINT GM$(I);                   │
620 NEXT I                          │
630 RETURN                          ┘
640 REM
650 COLOR 6                         ┐
```

2. ガンマンゲーム

```
660 FOR I=0 TO 9                 ┐
670 LOCATE X2,Y2+I               │自分のガンマンを出力するルーチン
680 PRINT GN$(I);                │
690 NEXT I                       │
700 RETURN                       ┘
710 REM
720 FOR I=0 TO 1                 ┐
730 SOUND 6,15:SOUND 7,7         │
740 SOUND 8,16:SOUND 9,16        │発射音
750 SOUND 10,16:SOUND 12,16      │
760 SOUND 13,0                   │
770 NEXT I                       ┘
780 I=30                                                   ┐
790 LOCATE I,Y2+5:PRINT" ";  ――弾を消す                    │
800 I=I-1                                                  │
810 P$=CHARACTER$(I,Y2+5)                                  │
820 IF P$="■" THEN 460        ――急所以外に当ったとき        │
830 IF P$="▦" THEN LOCATE I,Y2+5:PRINT" ";:GOTO 460―┐      │自分が弾を発射した
840 IF P$="□" THEN 1070       ――急所に当ったとき サボテンに当ったとき │ルーチン
850 IF I=1 THEN 460           ――端まで飛んだとき            │
860 COLOR 7                                                │
870 LOCATE I,Y2+5:PRINT"●";――弾を出力する                  │
880 GOTO 790                                               ┘
890 REM
900 FOR I=0 TO 1                 ┐
910 SOUND 6,15:SOUND 7,7         │
920 SOUND 8,16:SOUND 9,16        │発射音
930 SOUND 10,16:SOUND 12,16      │
940 SOUND 13,0                   │
950 NEXT I                       ┘
960 I=9                                                    ┐
970 LOCATE I,Y1+5:PRINT" "; ――弾を消す                     │
980 I=I+1                                                  │
990 P$=CHARACTER$(I,Y1+5)                                  │
1000 IF P$="■" THEN 530       ――急所以外に当ったとき        │敵が弾を発射した
1010 IF P$="▦" THEN LOCATE I,Y1+5:PRINT" ";:GOTO 530┐      │ルーチン
1020 IF P$="□" THEN 1220      ――急所に当ったとき サボテンに当ったとき │
1030 IF I=38 THEN 530         ――端まで飛んだとき            │
1040 COLOR 7                                               │
1050 LOCATE I,Y1+5:PRINT"●"; ――弾を出力する                │
1060 GOTO 970                                              ┘
1070 REM
1080 SOUND 7,&HF8                                          ┐
1090 SOUND 8,15:SOUND 9,15:SOUND 10,15                     │当ったときの音
1100 PLAY"RRRCCCRRCCCRRCCRCCCRRR#D#D#DRDDRDDDRCCRCCCR-B-BRCCCC ┘
1110 FOR I=Y1 TO Y1+3       ┐                ┐
1120 LOCATE X1,I            │頭を消す        │
1130 PRINT"          ";     │                │
1140 NEXT I                 ┘                │
1150 FOR I=0 TO 3           ┐                │敵の急所に当ったルーチン
1160 LOCATE X1,Y1-2+I       │頭を飛ばす      │
1170 PRINT GM$(I);          │                │
1180 NEXT I:RS=RS+1         ┘                │
1190 IF RS=5 THEN 1370 ――あなたの勝ち        ┘
1200 PAUSE 10
1210 GOTO 240
1220 REM
1230 PAUSE 10:SOUND 7,&HF8                                 ┐
1240 SOUND 8,15:SOUND 9,15:SOUND 10,15                     │当ったときの音
1250 PLAY"RRRCCCRRCCCRRCCRCCCRRR#D#D#DRDDRDDDRCCRCCCR-B-BRCCCC ┘
1260 FOR I=Y2 TO Y2+3       ┐                ┐
1270 LOCATE X2,I            │頭を消す        │
1280 PRINT"          ";     │                │
1290 NEXT I                 ┘                │
1300 FOR I=0 TO 3           ┐                │自分の急所に当ったルーチン
1310 LOCATE X2,Y2-2+I       │頭を飛ばす      │
1320 PRINT GN$(I);          │                │
1330 NEXT I:LS=LS+1         ┘                │
1340 IF LS=5 THEN 1510 ――あなたの負け        ┘
1350 PAUSE 10
1360 GOTO 240
1370 REM
1380 LOCATE  5,1:PRINT"LEFT SCORE=";LS            ┐
1390 LOCATE 20,1:PRINT"RIGHT SCORE=";RS           │
```

```
1400 LINE(0,15)-(39,21)," ",BF                         ┐
1410 LOCATE 0,16                                       │
1420 PRINT"              GAME OVER !!                  │
1430 PRINT                                             │
1440 PRINT"              ｱﾅﾀﾉ ｶﾁﾃﾞｽ｡                    ├ あなたの勝ち
1450 PRINT                                             │
1460 PRINT"           ﾓｳ1ﾄﾞ ﾔﾘﾏｽｶ? (Y/N)";              │
1470 I$=INPUT$(1)                                      │
1480 IF I$="Y" OR I$="y" THEN RUN                      │ ┐
1490 IF I$="N" OR I$="n" THEN CLS:END                  │ ├ 再ゲームか？
1500 GOTO 1470                                         ┘ ┘
1510 REM
1520 LOCATE  5,1:PRINT"LEFT SCORE=";LS                 ┐
1530 LOCATE 20,1:PRINT"RIGHT SCORE=";RS                │
1540 LINE(0,15)-(39,21)," ",BF                         │
1550 LOCATE 0,16                                       │
1560 PRINT"              GAME OVER !!                  │
1570 PRINT                                             ├ あなたの負け
1580 PRINT"               X1ﾉ ｶﾁﾃﾞｽ｡                    │
1590 PRINT                                             │
1600 PRINT"           ﾓｳ1ﾄﾞ ﾔﾘﾏｽｶ? (Y/N)";              │
1610 I$=INPUT$(1)                                      │
1620 IF I$="Y" OR I$="y" THEN RUN                      │
1630 IF I$="N" OR I$="n" THEN CLS:END                  ┘
1640 GOTO 1610
```

ゲーム終了．5対4でX１の勝ち！

3
キャラクターメーカー

X1はPCG（プログラマブル・キャラクター・ジェネレーター）機能を持っていて非常に綺麗なキャラクターを使えることができる反面，非常にデータを作るのが面倒です。通常このデータを作るには方眼紙にマス目を取り，そこに書き込んだものからドット単位で数値を計算して、それを2進に直す，という作業が必要です。これでは，せっかく良い機能が有るのに，使用するのにためらってしまいます。そこで，この作業をもう少し簡単にしようと思い作ってみましたのが，この“キャラクターメーカー”です。

プログラムの使用法

まず，プログラムを RUN させると，コマンドを尋ねてきます。コマンドは SAVE コマンドと EDIT コマンドの2種類しかありません。EDIT は，すきなキャラクターを作るための定義，SAVE は，この作ったキャラクターをデータとして保存するものです。コマンド別に使用法を書きますので，よく読んで下さい。

☆EDIT コマンド

コマンド待ちのときに“E”または“e”を押すと，このコマンドに入ります。すると画面中央に8×8のマス目が表示されますので，この中で自由にキャラクターを作って下さい。それでは使用キーですが，数字1～7までは，それに対応する色の変更です。スペースキーは空白を出力します。RETURN キーはキャラクターの定義に使います。カーソルキーはそのまま使えます。それ以外のキーを押せば，ドットが打てるようになっています。目的のキャラクターができましたら RETURN を押して下さい。すると，定義するキャラクターの番号を尋ねてきますので，00～1Fまでの番号を，16進で入力して下さい。暫くすると，キャラクターが定義され出力されます。修正したい場合ですが，続けてEコマンドを使用すれば前に定義したものが残っていますので，ご安心を……。

☆SAVE コマンド

コマンド待ちのときに“S”または“s”を押すと，このコマンドに入ります。すると，ファイルネームを尋ねてきますので，16文字以内で好きな名前を入力して下さい。後は，X1が自動的にプログラムに変換して出力してくれます。なお，ここで作ったデータは，X1で直接 LOAD することができるようになっています。ASCII 型式で SAVE されていますので，違うプログラムに，くっ付けることもできるようになっています。本誌の14番目の“ベジタブルくずし”のデータも，このキャラクターメーカーで作ったものです。

目的のプログラムに組み込む方法

Sコマンドで作られたデータは，ASCII 形式で出力されていますので，これを目的のプログラムの後に，MERGE コマンドを使って，くっ付けます。まず目的のプログラムを LOAD する。次にデータの入っているテープをセットして，MERGE [CR] とする。レコーダーが，カチャカチャ音を立てながら，ロードし，OKが出れば終了です。

これで目的のプログラムに，くっ付いた訳ですが，一応 LIST 60000－[CR] としてみて下さい。次に，必要のないデータを消去します。

データは，

60000 DEFCHR＄（—）＝HEXCHR＄（“——”）
＋HEXCHR＄（“——”）＋HEXCHR＄（“——”）

上記の形になっていますが，DEFCHR＄の中の数字が，キャラクターメーカーのEコマンドで使用したときのキャラクターナンバーに当ります。これを必要に合わせて変更すれば良い訳です。たとえば，スペースを変更したいときは，スペースのキャラクタコードは 32 ですので，DEFCHR ＄ の中の数字を 32 にすれば良いわけです。

最後に出力ですが，CGEN と PRINT＃0 を使って出力させれば終りです。

○キャラクターメーカーの利用法

では，もっと簡単に実際に例を上げてやってみましょう。

まず，プログラム1を打ち込んで走らせて見て下さい。〝●〞と〝○〞が40個ずつ表示されたと思います。この〝●〞と〝○〞を自分で定義してみることにしましょう。このプログラム1をカセットに SAVE しておきます。つぎに，キャラクターメーカーを LOAD RUN して下さい。定義するキャラクターは2個なので，好きなキャラクターを2個作り，No.0とNo.1に定義することにします。No.0とNo.1に定義し終えたら，新しいテープに，Sコマンドでデータを SAVE して下さい。これで，プログラム1とキャラクターのデータが入っているテープが2本できたはずです。

プログラム1

```
10 REM *** TEST PROGRAM ***   元のテストプログラム
20 '                          "●"と"○"を40個表示させる
30 '
40 FOR I=0 TO 39
50 PRINT CHR$(224);           "●"
60 NEXT I
65 PRINT
70 FOR I=0 TO 39
80 PRINT CHR$(225);           "○"
90 NEXT I
100 '
110 END
```

今度は，先ほど SAVE したプログラム1のテープを LOAD します。ここに，2度目に SAVE したキャラクターのデータを，付けるのですが，先ほど書いてあるように，キャラクターメーカーのSコマンドで出力されたテープは，ASCII 型式ですので，そのまま MERGE コマンドを使用することができるようになっています。データの入ったテープをセットして，MERGE CR として下さい。自動的にX1が1本のプログラムにしてくれます。そして，出来たのがプログラム2です。LIST を取ると分かるように不必要なデータが，沢山ありますので，DELETE コマンドで消去します。ここでは行番号，60020 ～60310 までですので，DELETE 60020 —60310 CR とします。

プログラム2

元のプログラムにキャラクターメーカーで作ったデータをくっつけたもの

```
10 REM *** TEST PROGRAM ***
20 '
30 '
40 FOR I=0 TO 39
50 PRINT CHR$(224);
60 NEXT I
65 PRINT
70 FOR I=0 TO 39
80 PRINT CHR$(225);
90 NEXT I
100 '
110 END
60000 DEF CHR$(&H0)=HEXCHR$("0000000000000000")+HEXCHR$("002C7EFFFFFF7E3C")+HEXC
HR$("1810000000000000")
60010 DEF CHR$(&H1)=HEXCHR$("00000000FF00FF00")+HEXCHR$("001818FFFFFFFFFF")+HEXC
HR$("0C0000FFFF00FFFF")
60020 DEF CHR$(&H2)=HEXCHR$("0000000000000000")+HEXCHR$("0000000000000000")+HEXC
HR$("0000000000000000")
60030 DEF CHR$(&H3)=HEXCHR$("0000000000000000")+HEXCHR$("0000000000000000")+HEXC
HR$("0000000000000000")

        ≀≀

60290 DEF CHR$(&H1D)=HEXCHR$("0000000000000000")+HEXCHR$("0000000000000000")+HEX
CHR$("0000000000000000")
60300 DEF CHR$(&H1E)=HEXCHR$("0000000000000000")+HEXCHR$("0000000000000000")+HEX
CHR$("0000000000000000")
60310 DEF CHR$(&H1F)=HEXCHR$("0000000000000000")+HEXCHR$("0000000000000000")+HEX
CHR$("0000000000000000")
```

ここが作ったデータ（60000～60010）

ここがいらないデータ（60020～60310）

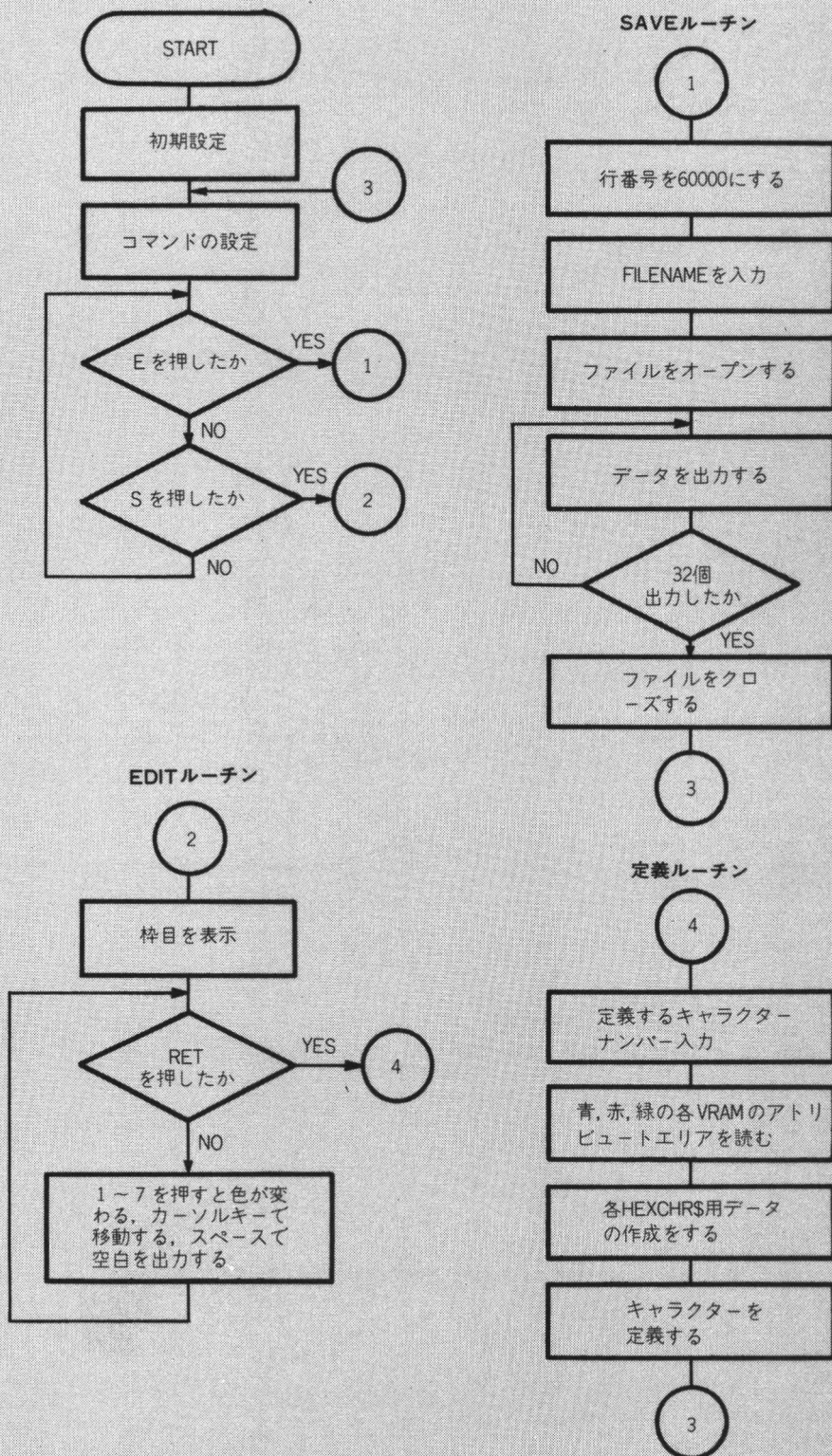
3．キャラクターメーカー

最後に DEF CHR＄ の中を必要に合わせて変更をします。ここでは，"●"＝224 "○"＝225ですので，行番号60000 の DEFCHR＄ の中味（&H0）を（224）に，行番号60010 の（&H1）を（225）とします。表示するキャラクターは RAM CG（キャラクタージェネレーター）ですので，プログラムの先頭に，CGEN1 を入れ最後に，CGEN0 としてやれば終りです。さあ，RUN してみて下さい。あなたの定義したキャラクターが出てきたはずです。プログラム3は完成し出来上ったものですので，参考にして下さい。

みなさんがオリジナルを作るときに注意することですが，CGEN1，CGEN0 を入れ忘れないこと，DEFCHR＄ の数を，間違がえないこと，などが上げられます。みなさんも，このキャラクターメーカーを使って，色々なオリジナルキャラクターを使ってみて下さい。

プログラム3

```
10 REM *** TEST PROGRAM ***       完成したプログラム
20 GOSUB 130 ——データを読み込む
30 CGEN 1 ————表示するキャラクターをRAMCGにする
40 FOR I=0 TO 39      }
50 PRINT CHR$(224);   }        └─コンピュータグラフィク(Computer Graphic)ではありません
60 NEXT I             }          キャラクタージェネレータ(Character Generator)のことです
70 PRINT———————改行
80 FOR I=0 TO 39      }
90 PRINT CHR$(225);   }
100 NEXT I            }
110 CGEN 0————表示するキャラクターをROMCGにする
120 END
130 DEF CHR$(224)=HEXCHR$("0000000000000000")+HEXCHR$("002C7EFFFFFF7E3C")+HEXCHR
$("1810000000000000")
140 DEF CHR$(225)=HEXCHR$("00000000FF00FF00")+HEXCHR$("001818FFFFFFFFFF")+HEXCHR
$("0C0000FFFF00FFFF")
150 RETURN
          └──ここに注目する!!                                   └RAM CG用の
                                                                  データ
```

キャラクターエディター・フローチャート
START
初期設定
3
コマンドの設定
Eを押したか
YES
1
NO
Sを押したか
YES
2
NO
SAVEルーチン
1
行番号を60000にする
FILENAMEを入力
ファイルをオープンする
データを出力する
NO
32個
出力したか
YES
ファイルをクローズする
3
EDITルーチン
2
枠目を表示
RET
を押したか
YES
4
NO
1～7を押すと色が変わる，カーソルキーで移動する，スペースで空白を出力する
定義ルーチン
4
定義するキャラクターナンバー入力
青，赤，緑の各VRAMのアトリビュートエリアを読む
各HEXCHR$用データの作成をする
キャラクターを定義する
3

3．キャラクターメーカー

```
1 REM*************************************
2 REM*                                   *
3 REM* ■■■ CHARACTER MAKER VER 1 ■■■ *
4 REM*                                   *
5 REM*      [C]COPYRIGHT 1983年 6月       *
6 REM*                                   *
7 REM*   FOR SHARP-X1 BY HEART SOFT      *
8 REM*                                   *
9 REM*************************************
100 CONSOLE 0,25:WIDTH 40
110 CLICK OFF:DIM C$(31,3)
120 FOR I=1 TO 3:FOR J=0 TO 31
130 C$(J,I)=STRING$(8,"0")
140 NEXT J:NEXT I
150 COLOR7:CLS
160 PRINT"■■■ SHARP-X1 CHARACTER MAKER VER 1.0 ■■■";
170 LOCATE 0,24
180 PRINT"■■■ COPYRIGHT 1983.6. BY HEART SOFT ■■■";
190 GOSUB 810
200 LOCATE 0,22:PRINT SPACE$(39);
210 LOCATE 0,22:COLOR 7
220 PRINT"      CHARACTER EDIT OR SAVE? (E/S)";
230 I$=INPUT$(1)
240 IF I$="E" OR I$="e" THEN 270
250 IF I$="S" OR I$="s" THEN 970
260 GOTO 230
270 REM ■■■ CHARACTER EDIT ■■■
280 LOCATE 15,10:PRINT" 12345678"
290 LOCATE 15,11:PRINT"1"
300 LOCATE 15,12:PRINT"2"
310 LOCATE 15,13:PRINT"3"
320 LOCATE 15,14:PRINT"4"
330 LOCATE 15,15:PRINT"5"
340 LOCATE 15,16:PRINT"6"
350 LOCATE 15,17:PRINT"7"
360 LOCATE 15,18:PRINT"8"
370 LOCATE 16,11
380 I$=INPUT$(1)
390 IF I$=CHR$(13) THEN 420
400 IF I$=CHR$(28) OR I$=CHR$(29) OR I$=CHR$(30) OR I$=CHR$(31) OR I$=" " PRINT
I$;:ELSE IF I$>"0" AND I$<"8" THEN COLOR VAL(I$):PRINT"■";
410 GOTO 380
420 LOCATE 0,22:PRINT SPACE$(39);
430 LOCATE 0,22:COLOR 7
440 INPUT"      CHARACTER NUMBER INPUT";N$
450 N=VAL("&H"+N$)
460 IF N<0 OR N>31 THEN 420
470 REM BLUE
480 C$(N,1)=""
490 FOR J=11 TO 18
500 X=1:P=0:I=0
510 FOR K=16 TO 24
520 IF CHARACTER$(K,J)=" "THEN 550
530 I=INP(&H2000+K+J*40)
540 IF I=1 OR I=5 OR I=3 OR I=7 THEN P=P+128/X
550 X=X*2:NEXT:C$(N,1)=C$(N,1)+RIGHT$("00"+HEX$(P),2)
560 NEXT
570 REM RED
580 C$(N,2)=""
590 FOR J=11 TO 18
600 X=1:P=0:I=0
610 FOR K=16 TO 24
620 IF CHARACTER$(K,J)=" "THEN 650
630 I=INP(&H2000+K+J*40)
640 IF I=2 OR I=3 OR I=6 OR I=7 THEN P=P+128/X
650 X=X*2:NEXT:C$(N,2)=C$(N,2)+RIGHT$("00"+HEX$(P),2)
660 NEXT
670 REM GREEN
680 C$(N,3)=""
690 FOR J=11 TO 18
700 X=1:P=0:I=0
710 FOR K=16 TO 24
720 IF CHARACTER$(K,J)=" "THEN 750
730 I=INP(&H2000+K+J*40)
```

[100–140] 変数の初期化

[220–250] コマンドの選択
E → EDIT
S → SAVE

[380–410] キー入力
カーソルコントロールキーはそのまま使用
1～7までの数字は色指定
スペースもそのまま使用
その他はドット指定

[420–460] 定義する番号の入力

[500] データ数値計算用

[530] VRAMのアトリビュートを読む

[540] 青・水色・シアン・白

[470–560] 青色の読み取り

[640] 赤・シアン・黄・白

[570–660] 赤色の読み取り

[670–] 緑色の読み取り

DEFCHR$(n)=HEXCHR$("——")+HEXCHR$("——")+HEXCHR$("——")

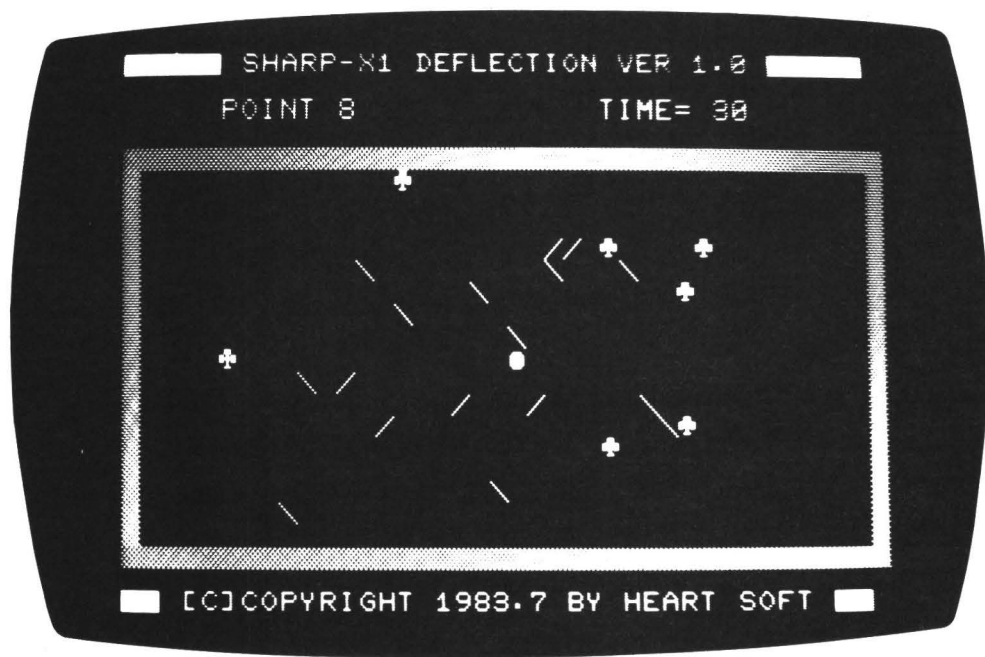
```
740 IF I=4 OR I=5 OR I=6 OR I=7 THEN P=P+128/X ────┬─緑・水色・黄・白
750 X=X*2:NEXT:C$(N,3)=C$(N,3)+RIGHT$("00"+HEX$(P),2)┘
760 NEXT
770 REM
780 DEF CHR$(N)=HEXCHR$(C$(N,1))+HEXCHR$(C$(N,2))+HEXCHR$(C$(N,3))── キャラクターの定義
790 GOSUB 810
800 GOTO 200
810 REM CHARACTER PRINT
820 LOCATE 5,3
830 PRINT"0 1 2 3 4 5 6 7 8 9 A B C D E F"
840 LOCATE 3,4
850 PRINT"0 ";
860 FOR I=0 TO 15
870 CGEN1:PRINT#0 CHR$(I);
880 CGEN0:PRINT" ";
890 NEXT I
900 LOCATE 3,6
910 PRINT"1 ";
920 FOR I=16 TO 31
930 CGEN1:PRINT#0 CHR$(I);
940 CGEN0:PRINT" ";
950 NEXT I
960 RETURN
970 REM DATA SAVE TO CMT
980 L!=60000!          ──────行番号
990 LOCATE 0,22:PRINT SPACE$(39);
1000 LOCATE 0,22:COLOR 7
1010 INPUT "      FILE NAME (16 STRING)";FL$
1020 OPEN"O",#1,FL$  ──────ファイルをオープンする
1030 FOR I=0 TO 31
1040 W$=STR$(L)+" DEF CHR$(&H"+HEX$(I)+")=HEXCHR$("+CHR$(34)+C$(I,1)+CHR$(34)+")
+HEXCHR$("+CHR$(34)+C$(I,2)+CHR$(34)+")+HEXCHR$("+CHR$(34)+C$(I,3)+CHR$(34)+")"
1050 PRINT#1,W$
1060 L=L+10:NEXT I:CLOSE────クローズする
1070 GOTO 200
```

（810〜960）32個の定義したキャラクターを表示させるルーチン

（980〜1070）SAVEルーチン

（1040）データ型式でSAVEしてあるが直接LOADすることができる

例　60000 DEFCHR$(&H1)＝HEXCHR$("――")＋HEXCHR$("――")
＋HEXCHR$("――")
このように出力される

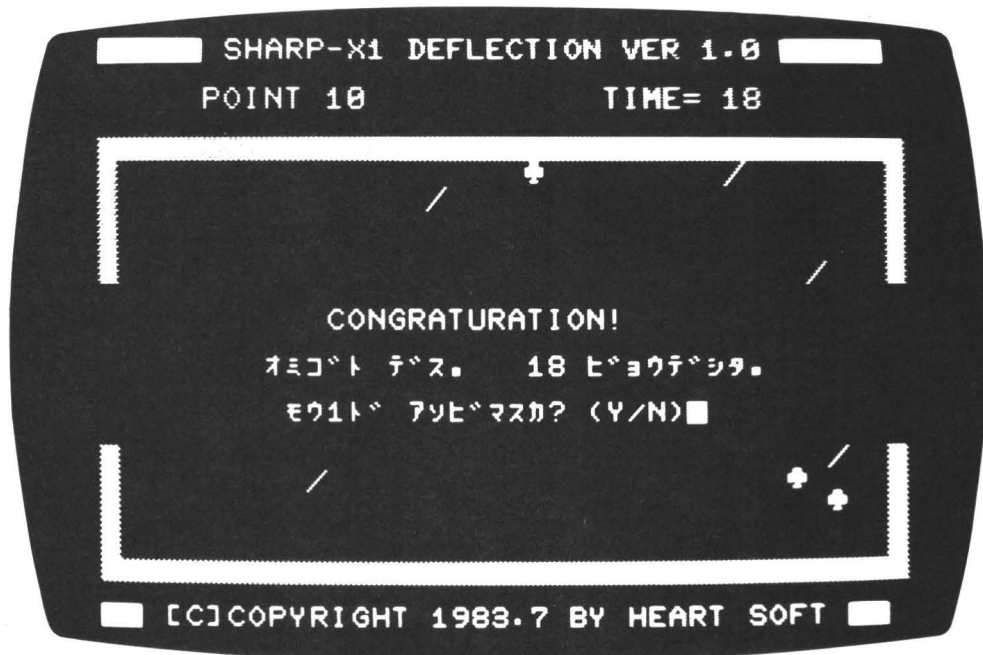
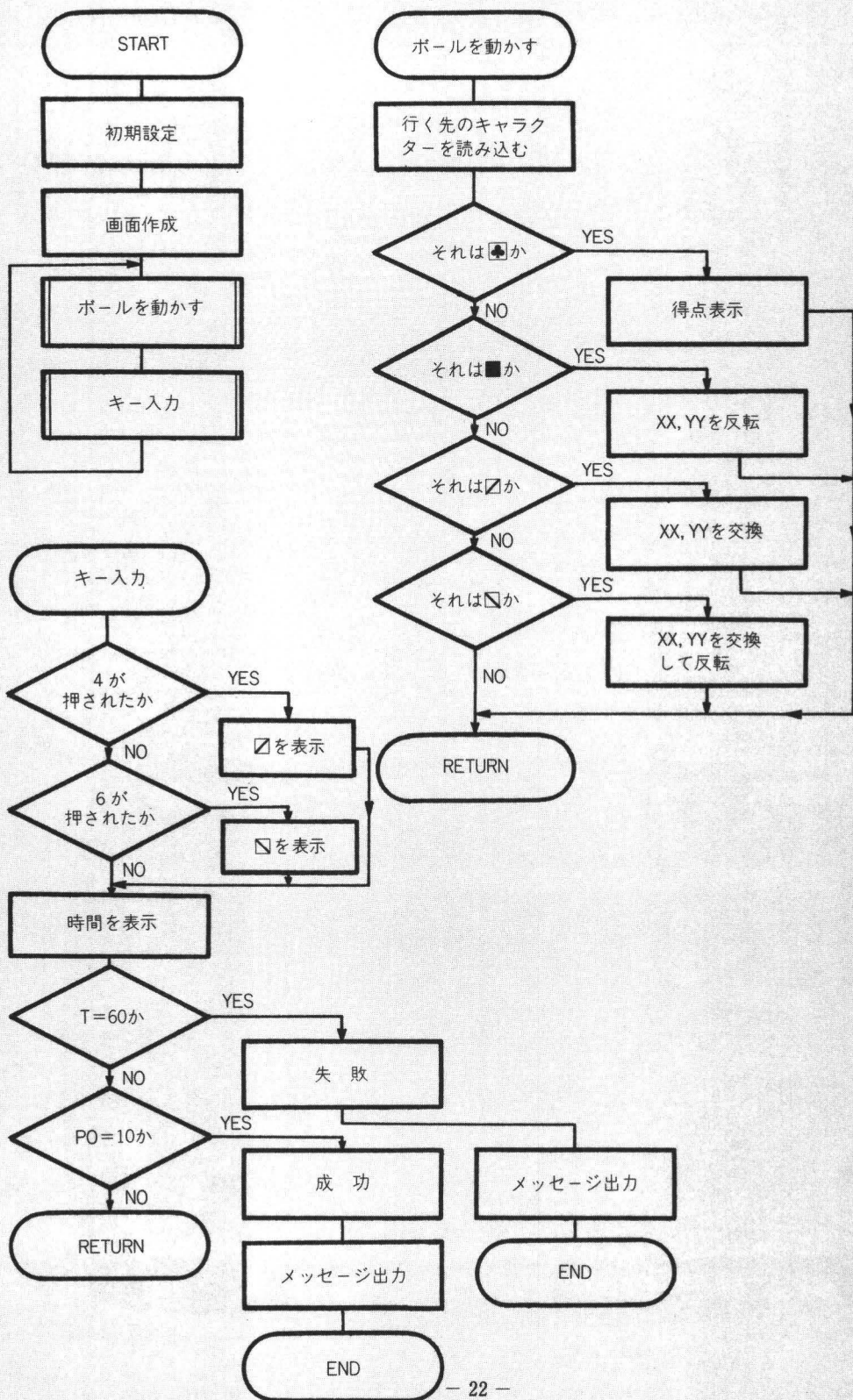
4 デフレクション

このゲームは私が生まれて初めて買った往年の名機，コモドール社のPET（読者の中で知っている人がいるでしょうか………？）にあったゲームで，シンプルですが，なかなか面白いゲームだったと思います。昔のことなので，これで良かったかどうか，よく覚えていないのですが，まあこのような感じだろうと思います。それと，一つだけ今でも分からないことなのですが，このゲームの名前です。デフレクション（DEFLECTION）を辞書で調べてみると〝屈折〟と載っています。何故，リフレクション〝反射〟ではなかったのでしょうか………？。

ゲームの遊び方

このゲームは動く玉をうまく反射板に当てて的に当てるゲームです。このゲームはタイム制限で，60秒の時間内に10コの的を取れるかを競うものです。使用するキーは4と6で，4を押すと〝／〟，6を押すと〝＼〟が出現します。どちらが，どっちへ行くかよく覚えておかないと，頭の中が混乱してしまいますので，考えてプレイして下さい。時間内ですべて取れるようになりましたら，次はどれくらい時間を短縮できるかに挑戦してみて下さい。

変数表

Ｘ，Ｙ→玉のＸ，Ｙ座標
ＸＸ，ＹＹ→玉の進行方向
Ｔ→時間
ＰＯ→ポイント
Ｉ＄→キー入力用
Ｐ＄→画面読み取り用

デフレクション・フローチャート
START
初期設定
画面作成
ボールを動かす
キー入力
ボールを動かす
行く先のキャラクターを読み込む
それは♣か
YES
得点表示
NO
それは■か
YES
XX, YYを反転
NO
それは◪か
YES
XX, YYを交換
NO
それは◩か
YES
XX, YYを交換して反転
NO
RETURN
キー入力
4が押されたか
YES
◪を表示
NO
6が押されたか
YES
◩を表示
NO
時間を表示
T＝60か
YES
失　敗
NO
PO＝10か
YES
成　功
メッセージ出力
NO
RETURN
メッセージ出力
END
END

```
1 REM※※※※※※※※※※※※※※※※※※※※※※※※※※※※※※※※※※※
2 REM※                                 ※
3 REM※   ■■■■ DEFLECTION VER 1.0 ■■■■   ※
4 REM※                                 ※
5 REM※      [C]Copyright 1983年 7月      ※
6 REM※                                 ※
7 REM※   FOR SHARP-X1 By Heart Soft    ※
8 REM※                                 ※
9 REM※※※※※※※※※※※※※※※※※※※※※※※※※※※※※※※※※※※
10 CONSOLE 0,25:WIDTH 40
20 CLS:CLICK OFF:COLOR 7,0:TIME=0
30 RANDOMIZE:XX=0:YY=0
40 LOCATE 0,0
50 PRINT"■■■■ SHARP-X1 DEFLECTION VER 1.0 ■■■■";
60 LOCATE 0,24
70 PRINT"■■ [C]COPYRIGHT 1983.7 BY HEART SOFT ■■";
80 LOCATE  5,2:PRINT "POINT= 0";
90 LOCATE 25,2:PRINT "TIME= 0";
100 COLOR 5
110 LINE(0,4)-(39,22),"▒",B
120 COLOR 6
130 FOR I=1 TO 15
140 X=INT(RND(1)*35)+2
150 Y=INT(RND(1)*16)+5
160 IF CHARACTER$(X,Y)<>" " THEN 140
170 LOCATE X,Y:PRINT"♣";
180 NEXT I
190 COLOR 2
200 X=INT(RND(1)*34)+3
210 Y=INT(RND(1)*15)+6
220 IF CHARACTER$(X,Y)<>" " THEN 200
230 LOCATE X,Y:PRINT"●";:PAUSE 10
240 LOCATE X,Y:PRINT" ";:PAUSE 10
250 LOCATE X,Y:PRINT"●";:PAUSE 10
260 XX=1:YY=0
270 LOCATE X,Y:PRINT" ";
280 P$=CHARACTER$(X+XX,Y+YY)
290 IF P$="♣" THEN GOSUB 460:GOSUB 840
300 IF P$="▒" THEN XX=-XX:YY=-YY:GOSUB 750
310 IF P$="/" THEN SWAP XX,YY:XX=-XX:YY=-YY:GOSUB 750
320 IF P$="\" THEN SWAP XX,YY
330 X=X+XX:Y=Y+YY:COLOR 2
340 LOCATE X,Y:PRINT"●";
350 IF X<2 OR X>37 OR Y<6 OR Y>20 THEN 400
360 COLOR 4
370 I$=INKEY$
380 IF I$="4" THEN IF CHARACTER$(X+XX,Y+YY)=" " THEN LOCATE X+XX,Y+YY:PRINT "/";
 ELSE 400
390 IF I$="6" THEN IF CHARACTER$(X+XX,Y+YY)=" " THEN LOCATE X+XX,Y+YY:PRINT "\";
 ELSE 400
400 GOSUB 430
410 IF T=60 THEN 500
420 GOTO 270
430 COLOR 7:T=TIME
440 LOCATE 30,2:PRINT T;
450 RETURN
460 COLOR 7:BEEP:PO=PO+1
470 IF PO=10 THEN 620
480 LOCATE 10,2:PRINT PO;
490 RETURN
500 REM
510 LINE(0,10)-(39,16)," ",BF
520 LOCATE 0,11
530 PRINT"                TIME OVER !!
540 LOCATE 0,13
550 PRINT"           ｼﾞｶﾝｷﾞﾚ ﾃﾞｽ｡ ";PO;"ｺ ﾄﾚﾏｼﾀ｡
560 LOCATE 0,15
570 PRINT"            ﾓｳ1ﾄﾞ ｱｿﾋﾞﾏｽｶ? (Y/N)";
580 I$=INPUT$(1)
590 IF I$="Y" OR I$="y" THEN RUN
600 IF I$="N" OR I$="n" THEN CLS:END
610 GOTO 580
620 REM
630 LOCATE 10,2:PRINT PO;
```

- 10～40：初期化ルーチン
- 50～90：画面表示
- 110：枠を表示する
- 120～180：的を15個ランダムに表示する
- 160, 220：スペースでなかったらやり直し
- 190～250：ボールを表示する
- 230～250：ボールを点滅させる
- 280～320：各処理ルーチン　的なら460へ行く　壁なら正反射させる　／又は＼なら反射させる
- 350：はみ出し処理
- 370～390：キー入力ルーチン　4，6で／と＼を表示する
- 410：タイムオーバー
- 430～450：時間表示ルーチンへ
- 470：10個取ったら成功
- 460～490：ポイント表示ルーチンへ
- 500～610：タイムオーバー
- 580～610：再ゲームか？

4．デフレクション

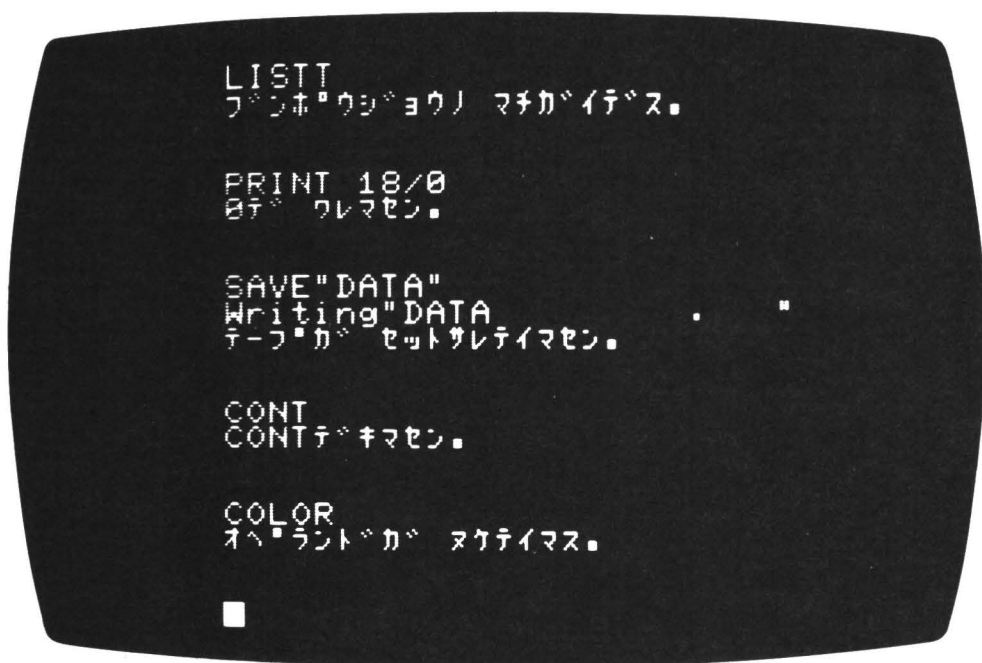
```
640 LINE(0,10)-(39,16)," ",BF
650 LOCATE 0,11
660 PRINT"            CONGRATURATION!
670 LOCATE 0,13
680 PRINT"         ｵﾐｺﾞﾄ ﾃﾞｽ｡   ";T;"ﾋﾞｮｳﾃﾞｼﾀ｡
690 LOCATE 0,15
700 PRINT"          ﾓｳ1ﾄﾞ ｱｿﾋﾞﾏｽｶ? (Y/N)";
710 I$=INPUT$(1)
720 IF I$="Y" OR I$="y" THEN RUN
730 IF I$="N" OR I$="n" THEN CLS:END
740 GOTO 710
750 REM ﾊﾝｼｬ ｵﾝ
760 SOUND 1,1  :SOUND 0,10
770 SOUND 3,1  :SOUND 2,10
780 SOUND 5,1  :SOUND 4,10
790 SOUND 6,31 :SOUND 7,&HF8
800 SOUND 8,16 :SOUND 9,16
810 SOUND 10,16:SOUND 11,20
820 SOUND 12,15:SOUND 13,0
830 RETURN
840 REM ﾄｸﾃﾝ ｵﾝ
850 SOUND 1,0  :SOUND 0,90
860 SOUND 7,&HFE:SOUND 13,0
870 RETURN
```

（640〜730：成功／）

5 日本語エラーメッセージ

X 1は強力な BASIC が搭載されていて非常に使いやすいのですが，たくさんの命令が有るかわりに，それだけエラーメッセージも多く，何と 50 個もあります。これをすべて覚えるのは容易ではありません。BASIC は，もともと米国生まれですので，命令は英語ですし，もち論エラーメッセージも英語です。これではなかなかわかりにくく，いちいちエラーメッセージ表を手に………。というふうになります。そこで，よりわかりやすくするために，エラーメッセージをすべて日本語にしてみました。それが，この日本語エラーメッセージです。これで安心してエラーが出せる………かな？

プログラムの改造法

メッセージはすべて DATA 文にしてありますので，自由に書き直してみて下さい。ただし，エラーメッセージの入る番地は，＆H736A～＆H7697（814文字）までです。これをオーバーしないように，くれぐれも注意して下さい。もし，オーバーしますと，まず間違いなく暴走します。DATA 文中に出てくる〝＿〟は，そこにエラーメッセージが指定されていないパスの標です。

日本語エラーメッセージの例

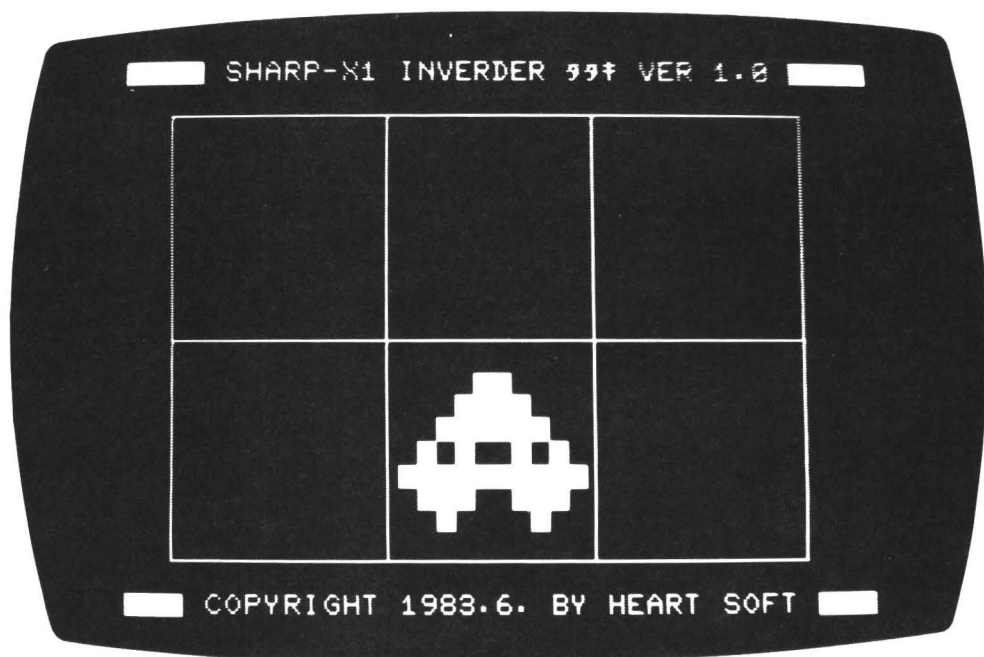
```
1 REM***************************************
2 REM*                                     *
3 REM*  ■■■■ Japanese Error Mesage ■■■■   *
4 REM*                                     *
5 REM*      (C)Copyright 1983年 7月        *
6 REM*                                     *
7 REM*   For SHARP-X1 BY Heart Soft        *
8 REM*                                     *
9 REM***************************************
10 WIDTH 40:CLS
20 PRINT"*** Japanese Error Message Ver 1.0 ***";
30 LOCATE 0,24
40 PRINT"*** (C)Copyright 1983 By Heart Soft ***";
50 LOCATE 7,14
60 PRINT"ﾀﾀﾞｲﾏ ﾃﾞｰﾀﾉ ﾖﾐｺﾐﾁｭｳﾃﾞｽ｡"
70 LOCATE 10,16
80 PRINT"ｼﾞﾊﾞﾗｸ ｵﾏﾁｸﾀﾞｻｲ｡"
90 FOR I=&H736A TO &H7697
100 POKE I,0:NEXT I
110 AD=&H736A:FOR I=1 TO 50:READ I$
120 FOR J=1 TO LEN(I$)
130 POKE AD+J,ASC(MID$(I$,J,1))
140 NEXT J:POKE AD+J,0
150 AD=AD+J:NEXT I
160 REM
170 DATA"FORｶﾞｱﾘﾏｾﾝ｡"
180 DATA"ﾌﾞﾝﾎﾟｳｼﾞｮｳﾉ ﾏﾁｶﾞｲﾃﾞｽ｡"
190 DATA"GOSUBｶﾞｱﾘﾏｾﾝ｡"
200 DATA"DATAｶﾞ ﾀﾘﾏｾﾝ｡"
210 DATA"ｶﾝｽｳﾃﾞﾊ ｱﾘﾏｾﾝ｡"
220 DATA"ｷｮﾖｳﾊﾝｲｦ ｺｴﾏｼﾀ｡"
230 DATA"ﾒﾓﾘｰｶﾞ ﾀﾘﾏｾﾝ｡"
240 DATA"ﾄﾋﾞｻｷｶﾞ ｱﾘﾏｾﾝ｡"
250 DATA"DIMﾃﾞ ｼﾃｲｻﾚﾃｲﾏｾﾝ｡"
260 DATA"DIMｶﾞ 2ｶｲｱﾘﾏｽ｡"
270 DATA"0ﾃﾞ ﾜﾚﾏｾﾝ｡"
280 DATA"ﾁｮｸｾﾂ ｼﾞｯｺｳ ﾃﾞｷﾏｾﾝ｡"
290 DATA"ﾓｼﾞﾄ ｽｳｼﾞﾊ ｲｯｼｮﾆ ﾃﾞｷﾏｾﾝ｡"
300 DATA"_255ｦ ｺｴﾏｼﾀ｡"
310 DATA"ｼｷｶﾞ ﾌｸｻﾞﾂ ｽｷﾞﾏｽ｡"
320 DATA"CONTﾃﾞｷﾏｾﾝ｡"
330 DATA"ﾃｲｷﾞｻﾚﾃ ｲﾏｾﾝ｡"
340 DATA"RESUMEｶﾞ ｱﾘﾏｾﾝ｡"
350 DATA"ERRORｶﾞ ｱﾘﾏｾﾝ｡"
360 DATA"ﾃｲｷﾞ ｻﾚﾃｲﾅｲﾏｾﾝ｡"
370 DATA"ｵﾍﾟﾗﾝﾄﾞｶﾞ ﾇｹﾃｲﾏｽ｡"
380 DATA"ｷﾞｮｳｶﾞ ﾅｶﾞｽｷﾞﾏｽ｡"
390 DATA"_ｶﾞﾒﾝﾃﾞﾊ ｱﾘﾏｾﾝ｡"
400 DATA"REPEATｶﾞ ｱﾘﾏｽﾝ｡"
410 DATA"ﾃｰﾌﾟｶﾞ ｾｯﾄｻﾚﾃｲﾏｾﾝ｡"
420 DATA"_ﾃﾟｰﾌﾟｶﾞ ﾖﾒﾏｾﾝ｡"
430 DATA"ｹｲｼｷｶﾞ ﾁｶﾞｲﾏｽ｡"
440 DATA"ﾅﾆﾓﾊｲｯﾃ ｲﾏｾﾝ｡"
450 DATA"WENDｶﾞ ｱﾘﾏｾﾝ｡"
460 DATA"WHILEｶﾞ ｱﾘﾏｾﾝ｡"
470 DATA"DISK BASICﾖｳ ﾃﾞｽ｡"
480 DATA"NEXTｶﾞｱﾘﾏｾﾝ｡"
490 DATA"ﾅｶﾞｽｷﾞﾏｽ｡"
500 DATA"UNTILｶﾞｱﾘﾏｾﾝ｡"
510 DATA"255ｦ ｺｴﾃｲﾏｽ｡"
520 DATA"ｼｮｳﾁｭｳ ﾃﾞｽ｡"
530 DATA"ﾅﾝﾊﾞｰｶﾞ ｵｶｼｲ｡"
540 DATA"ﾌｧｲﾙｶﾞ ｱﾘﾏｾﾝ｡"
550 DATA"ｵｰﾌﾟﾝ ｼﾃｲﾏｽ｡"
560 DATA"_ﾆｭｳｼｭﾂﾘｮｸ ｴﾗｰ ﾃﾞｽ｡"
570 DATA"ﾍﾝｺｳｻﾚﾃ ｲﾏｽ｡"
580 DATA"___ﾊｲﾘｷﾘﾏｾﾝ｡"
590 DATA"ｵﾜﾘｶﾞ ｱﾘﾏｾﾝ｡"
600 DATA"___FATｶﾞ ｺﾜﾚﾃｲﾏｽ｡"
610 DATA"ﾃﾞｨｽｸﾘﾌﾟﾀｶﾞ ﾁｶﾞｲﾏｽ｡"
620 DATA"ﾊﾞﾝｺﾞｳｶﾞ ﾁｶﾞｲﾏｽ｡"
630 DATA"ﾊﾟｽﾜｰﾄﾞｶﾞ ﾁｶﾞｲﾏｽ｡"
640 DATA"____ｵｰﾌﾟﾝ ｻﾚﾃｲﾏｾﾝ｡"
650 DATA"ﾂﾒ ｶﾞ ｵﾚﾃｲﾏｽ｡"
660 DATA"ｷｶｲｶﾞ ﾂﾅｶﾞｯﾃ ｲﾏｾﾝ｡"
```

（80〜100行）元のエラーメッセージをクリアーする

（110〜150行）50個のデータを読み込み ASCIIコードに直してメモリーに入れる

（170行以降）以下エラーメッセージのデータ 全部で50個

（300行の _ ）_は，その番号にエラーメッセージが定義されていないことを表す．

6
インベーダーたたき

インベーダーゲーム，これを読者の中で知らない人は，まずいないでしょう。モグラたたき，これも知らない人は少ないと思います。どちらかをX1で作ろうと思ったのですが，インベーダーを作るのは，ちょっと難しいので，モグラたたきにすることにしました。しかし，インベーダーも考えていたことですし，この2つを合体させてしまったのが，このインベーダーたたきです。初めはキー入力のところに，INKEY$ を使ってみたのですが，押し続けたりすると，リピートが利いてしまい，実用にはなりませんでした。そこで考えて，STICK 文を使うことにしました。これは元々ジョイスティック用のコマンドなのですが，代りにテンキーを使うこともできるようになっています。

ゲームの遊び方

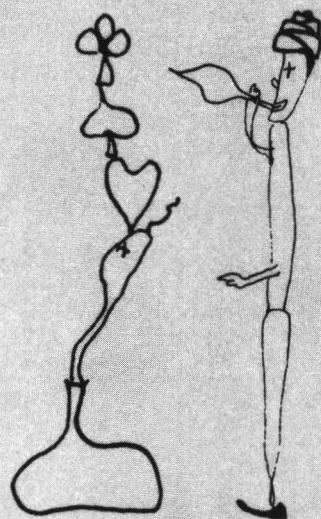
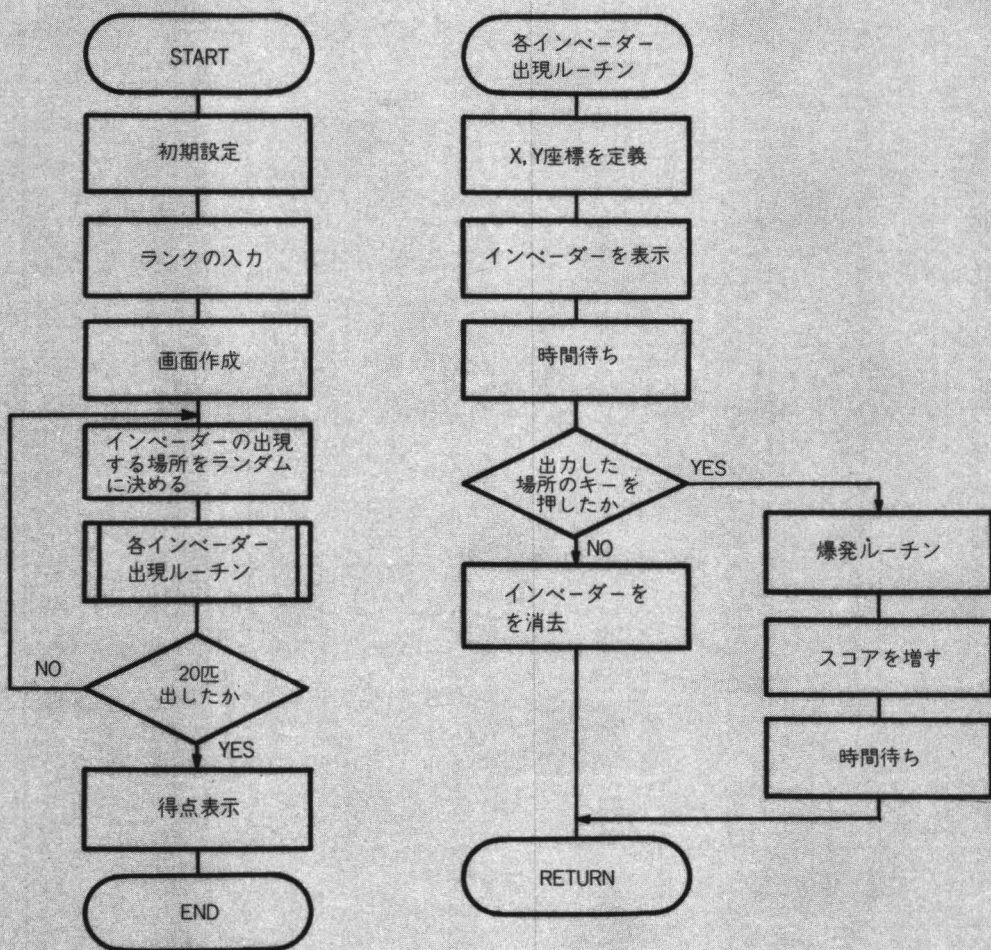
6つの升の中に次々とインベーダーが出現します。これを素早く叩き落とすゲームです。操作キーはテンキーの1～6を使い，左下から右下の3つは，1～3で，左上から右上までの3つが，4～6です。得点ですがインベーダーは，10点～30点までの3種数，これは本物のインベーダーゲームの点数と同じです。ＵＦＯは，100点～300点までのランダム得点です。これらが全部バラバラに出現し，全部で20回出てきます。ランクが1～3まであり，ランク2までは楽に打てると思いますが，ランク1になると少し大変かも知れません。うまくなれば，1000点の大台に乗ることも可能ですが，あまり興奮してキーをこわさないようにして下さい。

プログラムの改造法

インベーダーやＵＦＯ，爆発のデータは直接PRINTで書いてありますので，ここをギャラクシアンにすれば，ギャラクシアンたたきになりますし，ハエにすれば，ハエたたきになります。自分の好きなキャラクターにして，オリジナルな○○たたきにしてみて下さい。

変数表

P→プレイヤーのランク
Z→インベーダーの出る場所
ＩＮＶ→インベーダーの種類
Ｉ→キー入力用
Ｘ，Ｙ→インベーダーが出現するＸ，Ｙ座標

インベーダーたたき・フローチャート
START
初期設定
ランクの入力
画面作成
インベーダーの出現する場所をランダムに決める
各インベーダー出現ルーチン
NO
20匹出したか
YES
得点表示
END
各インベーダー出現ルーチン
X, Y座標を定義
インベーダーを表示
時間待ち
出力した場所のキーを押したか
YES
NO
インベーダーをを消去
爆発ルーチン
スコアを増す
時間待ち
RETURN

```
1 REM*************************************
2 REM*                                   *
3 REM* ███ INVERDER ﾀﾀｷ VER 1.0 ███      *
4 REM*                                   *
5 REM*      [C]COPYRIGHT 1983年 6月       *
6 REM*                                   *
7 REM*  FOR SHARP-X1 BY HEART SOFT       *
8 REM*                                   *
9 REM*************************************
10 WIDTH 40:CONSOLE 0,25
20 CLICK OFF:CLS:SC=0:COLOR 7,0
30 PRINT"████ SHARP-X1 INVERDER ﾀﾀｷ VER 1.0 ████";
40 LOCATE 0,24
50 PRINT"███ COPYRIGHT 1983.6. BY HEART SOFT ███";
60 LOCATE 12,10:PRINT"1.....ｼﾞｮｳｷｭｳ ｼｬ";
70 LOCATE 12,12:PRINT"2.....ﾁｭｳ ｷｭｳ ｼｬ";
80 LOCATE 12,14:PRINT"3.....ｼｮ  ｷｭｳ ｼｬ";
90 LOCATE 10,20
100 PRINT"ﾄﾞﾉ ﾗﾝｸﾆ ｼﾏｽｶ? (1-3)";
110 I$=INPUT$(1)
120 IF I$<"1" OR I$>"3" THEN 100
130 P=VAL(I$)
140 GOSUB 1520
150 FOR J=1 TO 20
160 Z=INT(RND(1)*6)+1
170 ON Z GOSUB 200,290,380,470,560,650
180 NEXT J
190 GOTO 1800
200 REM
210 X=3:Y=3
220 INV=INT(RND(1)*4)+1
230 ON INV GOSUB 750,870,990,1110
240 PAUSE P*5
250 I=STICK(0)
260 IF I=4 THEN GOSUB 1230
270 GOSUB 1410
280 RETURN
290 REM
300 X=14:Y=3
310 INV=INT(RND(1)*4)+1
320 ON INV GOSUB 750,870,990,1110
330 PAUSE P*5
340 I=STICK(0)
350 IF I=5 THEN GOSUB 1230
360 GOSUB 1410
370 RETURN
380 REM
390 X=25:Y=3
400 INV=INT(RND(1)*4)+1
410 ON INV GOSUB 750,870,990,1110
420 PAUSE P*5
430 I=STICK(0)
440 IF I=6 THEN GOSUB 1230
450 GOSUB 1410
460 RETURN
470 REM
480 X=3:Y=13
490 INV=INT(RND(1)*4)+1
500 ON INV GOSUB 750,870,990,1110
510 PAUSE P*5
520 I=STICK(0)
530 IF I=1 THEN GOSUB 1230
540 GOSUB 1410
550 RETURN
560 REM
570 X=14:Y=13
580 INV=INT(RND(1)*4)+1
590 ON INV GOSUB 750,870,990,1110
600 PAUSE P*5
610 I=STICK(0)
620 IF I=2 THEN GOSUB 1230
630 GOSUB 1410
640 RETURN
650 REM
```

- 100〜130：ランクの入力，1〜3まで
- 140：画面の枠を作る
- 150〜190：20匹のインベーダーを表示する
- 200〜280：No. 1
  - 220：インベーダーの種類を決める
  - 240：ランクに合わせて時間待ち
  - 260：爆発ルーチンへ
  - 270：消去ルーチンへ
- 290〜370：No. 2
- 380〜460：No. 3
- 470〜550：No. 4
- 560〜640：No. 5

Z＝1……左下
　2……中下
　3……右下
　4……左上
　5……中上
　6……右上

INV＝1……10点インベーダー
　　2……20点　〃
　　3……30点　〃
　　4……UFO 100点〜300点

6．インベーダーたたき

```
660 X=25:Y=13
670 INV=INT(RND(1)*4)+1
680 ON INV GOSUB 750,870,990,1110
690 PAUSE P*5
700 I=STICK(0)
710 IF I=3 THEN GOSUB 1230
720 GOSUB 1410
730 RETURN
740 END
750 REM
760 COLOR 5
770 LOCATE X,Y  :PRINT"   ████   "
780 LOCATE X,Y+1:PRINT" ████████ "
790 LOCATE X,Y+2:PRINT"██████████"
800 LOCATE X,Y+3:PRINT"██  ██  ██"
810 LOCATE X,Y+4:PRINT"██████████"
820 LOCATE X,Y+5:PRINT"██████████"
830 LOCATE X,Y+6:PRINT"  ██  ██  "
840 LOCATE X,Y+7:PRINT" █  ██  █ "
850 LOCATE X,Y+8:PRINT"  █    █  "
860 RETURN
870 REM
880 COLOR 6
890 LOCATE X,Y  :PRINT" █      █ "
900 LOCATE X,Y+1:PRINT"  █    █  "
910 LOCATE X,Y+2:PRINT"   █  █   "
920 LOCATE X,Y+3:PRINT"  ██████  "
930 LOCATE X,Y+4:PRINT" ██ ██ ██ "
940 LOCATE X,Y+5:PRINT"██████████"
950 LOCATE X,Y+6:PRINT"█ ██████ █"
960 LOCATE X,Y+7:PRINT"█ █    █ █"
970 LOCATE X,Y+8:PRINT"   █  █   "
980 RETURN
990 REM
1000 COLOR 4
1010 LOCATE X,Y  :PRINT"          "
1020 LOCATE X,Y+1:PRINT"    ██    "
1030 LOCATE X,Y+2:PRINT"   ████   "
1040 LOCATE X,Y+3:PRINT"  ██████  "
1050 LOCATE X,Y+4:PRINT" ██ ██ ██ "
1060 LOCATE X,Y+5:PRINT" ████████ "
1070 LOCATE X,Y+6:PRINT"  █ ██ █  "
1080 LOCATE X,Y+7:PRINT" █      █ "
1090 LOCATE X,Y+8:PRINT"  █    █  "
1100 RETURN
1110 REM
1120 COLOR 3
1130 LOCATE X,Y  :PRINT"          "
1140 LOCATE X,Y+1:PRINT"    ██    "
1150 LOCATE X,Y+2:PRINT"   ████   "
1160 LOCATE X,Y+3:PRINT"  ██████  "
1170 LOCATE X,Y+4:PRINT" █ █  █ █ "
1180 LOCATE X,Y+5:PRINT"██████████"
1190 LOCATE X,Y+6:PRINT" ███  ███ "
1200 LOCATE X,Y+7:PRINT"  █    █  "
1210 LOCATE X,Y+8:PRINT"          "
1220 RETURN
1230 REM
1240 COLOR 2
1250 FOR K=1 TO 10:BEEP1:FOR L=1 TO 10:NEXT L:BEEP 0:NEXT K
1260 LOCATE X,Y  :PRINT"  █    █  "
1270 LOCATE X,Y+1:PRINT"█  █  █  █"
1280 LOCATE X,Y+2:PRINT" █  ██  █ "
1290 LOCATE X,Y+3:PRINT"  █    █  "
1300 LOCATE X,Y+4:PRINT"██      ██"
1310 LOCATE X,Y+5:PRINT"  █    █  "
1320 LOCATE X,Y+6:PRINT" █  ██  █ "
1330 LOCATE X,Y+7:PRINT"█  █  █  █"
1340 LOCATE X,Y+8:PRINT"  █    █  "
1350 IF INV=1 THEN SC=SC+10
1360 IF INV=2 THEN SC=SC+20
1370 IF INV=3 THEN SC=SC+30
1380 IF INV=4 THEN SC=SC+INT(RND(1)*30)*10+10
1390 PAUSE 10
1400 RETURN
```

- 660〜730：No. 6
- 770〜850：10点インベーダー
- 890〜970：20点インベーダー
- 1010〜1090：30点インベーダー
- 1130〜1210：UFO
- 1250：爆発音
- 1260〜1340：爆発
- 1350〜1380：出て来たインベーダーによってスコアを加算する
- RND(1)*30)*10+10：100点〜300点の間の乱数(UFO)

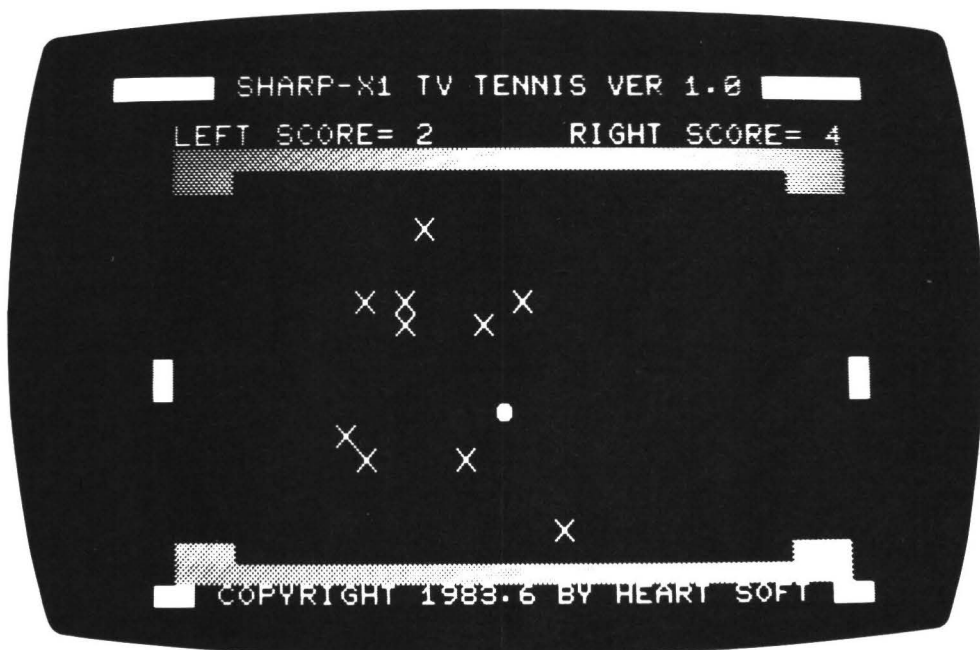
```
1410 REM
1420 LOCATE X,Y  :PRINT"                  "
1430 LOCATE X,Y+1:PRINT"                  "
1440 LOCATE X,Y+2:PRINT"                  "
1450 LOCATE X,Y+3:PRINT"                  "
1460 LOCATE X,Y+4:PRINT"                  "
1470 LOCATE X,Y+5:PRINT"                  "
1480 LOCATE X,Y+6:PRINT"                  "
1490 LOCATE X,Y+7:PRINT"                  "
1500 LOCATE X,Y+8:PRINT"                  "
1510 RETURN
1520 REM
1530 CLS:COLOR 7,0
1540 PRINT"■■■■ SHARP-X1 INVERDER ﾀﾀｷ VER 1.0 ■■■■";
1550 LOCATE 0,24
1560 PRINT"■■■ COPYRIGHT 1983.6. BY HEART SOFT ■■■";
1570 LOCATE 0,2
1580 PRINT"  ┌──────────┬──────────┬──────────┐
1590 PRINT"  │          │          │          │
1600 PRINT"  │          │          │          │
1610 PRINT"  │          │          │          │
1620 PRINT"  │          │          │          │
1630 PRINT"  │          │          │          │
1640 PRINT"  │          │          │          │
1650 PRINT"  │          │          │          │
1660 PRINT"  │          │          │          │
1670 PRINT"  │          │          │          │
1680 PRINT"  ├──────────┼──────────┼──────────┤
1690 PRINT"  │          │          │          │
1700 PRINT"  │          │          │          │
1710 PRINT"  │          │          │          │
1720 PRINT"  │          │          │          │
1730 PRINT"  │          │          │          │
1740 PRINT"  │          │          │          │
1750 PRINT"  │          │          │          │
1760 PRINT"  │          │          │          │
1770 PRINT"  │          │          │          │
1780 PRINT"  └──────────┴──────────┴──────────┘
1790 RETURN
1800 REM
1810 COLOR 7,0
1820 LINE(0,10)-(39,19)," ",BF
1830 LOCATE 0,12
1840 PRINT"              ■■■ GAME OVER ■■■"
1850 LOCATE 0,14
1860 PRINT"          ｱﾅﾀﾉ ﾄｸﾃﾝﾊ ";SC;"ﾃﾝ ﾃﾞｼﾀ"
1870 LOCATE 0,18
1880 PRINT"            ﾓｳ1ﾄﾞ ﾔﾘﾏｽｶ? (Y/N)";
1890 I$=INPUT$(1)
1900 IF I$="Y" OR I$="y" THEN RUN
1910 IF I$="N" OR I$="n" THEN CLS:END
1920 GOTO 1890
```

1420～1500：消去

1830～1920：ゲームオーバー時
Y→RUN
N→END

画面No.

No. 4	No. 5	No. 6
No. 1	No. 2	No. 3

7
TVテニス

近頃，ゲームセンターには数多くのゲームが，所狭しと置いてあり，中にはＣＰＵを，４つも持ち，大変大掛りなものも多いのですが，なんといってもＴＶゲームの本家は，このようなＴＶテニスが原形です。これは，インベーダーや，ブロックくずし等が出る前のもので，白黒ディスプレイに，パドルが２つあり両側２人で玉をただ打ち返すものでした。しかし，それが出た当時は､「家庭用テレビでテニスができる！」と，驚いたものでした。私も昔，大枚３万円もはたいて，買った思い出があります。そこで，本機を使ってＴＶテニスを作ってみようと思ったのですが，今更ただそのまま再現したのでは，あまりにつまらないので，途中に，×を出し乱数的な要素も組み込みました。

ゲームの遊び方

これは今説明した通り，動く玉をパドルに当てて相手の方へ打ち返し得点を競うものです。右側の人はテンキーの，9と3で，左側の人は，7と1で動かします。パドルや壁に当ったときは正反射しますが，×に当ったときはランダムに跳ね返るようになっています。どちらかが5ポイント取ると，勝負が決まります。

変数表

Y 1→右ラケットのY座標
Y 2→左ラケットのY座標
I 1 $→右用のキー入力
I 2 $→左用のキー入力
X 3→BALLのX座標
Y 3→BALLのY座標
C，D→乱数（1 or －1）
L，S→左のスコア
R，S→右のスコア
A，B→ラケットを動かすときの代用変数

7．TVテニス

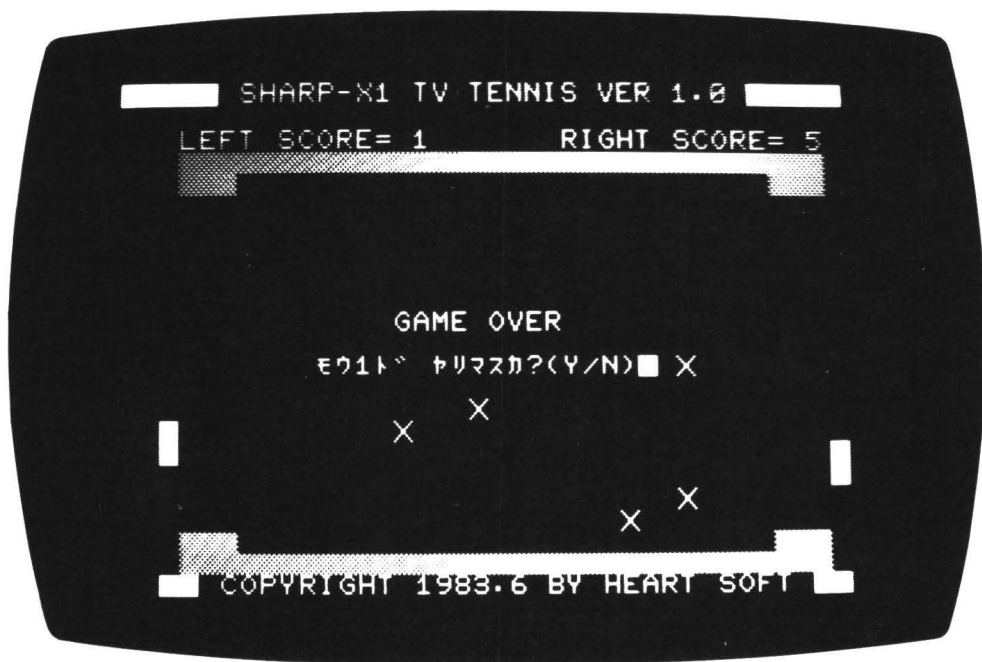
TVテニス・フローチャート
START
初期設定
画面作成
ボールの出る所決定
ラケット(R)
ラケット(L)
ボールを動かす
ボールがラケットからはずれたか
NO
YES
得点加算
どちらかが5点か
NO
YES
ゲームオーバー
END
ボールの出る所決定
X, Y座標をランダムに決定
そこはスペースか
NO
YES
進む方向をランダムにセット
RETURN
ボールを動かす
それは▩か
YES
NO
それは■か
YES
NO
ボールの進行方向を正反射にセット
それは⊠か
NO
YES
ボールの進行方向をランダムにセット
ボール移動
RETURN
ラケット(R)
9が押されたか
NO
YES
3が押されたか
NO
YES
A=−1
A=1
上はじか下はじか
YES
NO
ラケット移動
RETURN
ラケット(L)
7が押されたか
NO
YES
1が押されたか
NO
YES
B=−1
B=1
上はじか下はじか
YES
NO
ラケット移動
RETURN

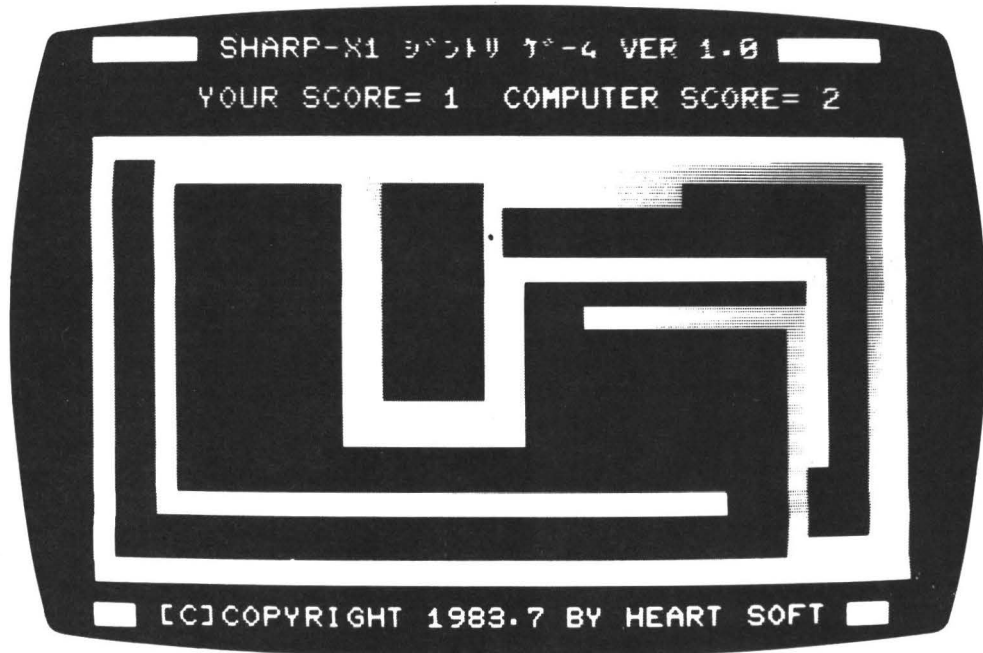
```
1 REM*************************************
2 REM*                                   *
3 REM* ■■■■ TV TENNIS VER. 1.0 ■■■■    *
4 REM*                                   *
5 REM*      [C]COPYRIGHT 1983年 6月      *
6 REM*                                   *
7 REM*  FOR SHARP-X1 BY HEART SOFT       *
8 REM*                                   *
9 REM*************************************
10 WIDTH 40:CONSOLE 0,25
20 CLICK OFF:COLOR 7,0:CLS
30 Y1=12:Y2=12
40 GOSUB 620   :'ｶﾞﾒﾝ ｾｯﾃｲ
50 GOSUB 110   :'ﾎﾞｰﾙ ｽﾀｰﾄ
60 GOSUB 260   :'ﾐｷﾞ ﾗｹｯﾄ
70 GOSUB 340   :'ﾋﾀﾞﾘ ﾗｹｯﾄ
80 GOSUB 170   :'ﾎﾞｰﾙ
90 IF X3<=1 OR X3=>38 THEN 420
100 GOTO 60
110 REM
120 X3=INT(RND(1)*20)+10
130 Y3=INT(RND(1)*14)+7
140 IF CHARACTER$(X3,Y3)<>" " THEN 120
150 GOSUB 820
160 RETURN
170 REM
180 LOCATE X3,Y3:PRINT" ";
190 P$=CHARACTER$(X3+C,Y3+D)
200 IF P$="▓" THEN D=-D
210 IF P$="■" THEN C=-C
220 IF P$="X" THEN GOSUB 820
230 X3=X3+C:Y3=Y3+D
240 COLOR 6
250 LOCATE X3,Y3:PRINT"●";
260 REM
270 I1=STICK(0):COLOR 2
280 IF I1= 9  THEN A=-1
290 IF I1= 3  THEN A=1
300 IF Y1+A<3 OR Y1+A>20 THEN RETURN
310 Y1=Y1+A:A=0
320 LOCATE 37,Y1:PRINT" ";CHR$(29,31);"■";CHR$(29,31);"■";CHR$(29,31);" ";
330 RETURN
340 REM
350 I2=STICK(0):COLOR 4
360 IF I2= 7  THEN B=-1
370 IF I2= 1  THEN B=1
380 IF Y2+B<3 OR Y2+B>20 THEN RETURN
390 Y2=Y2+B:B=0
400 LOCATE 2,Y2:PRINT" ";CHR$(29,31);"■";CHR$(29,31);"■";CHR$(29,31);" ";
410 RETURN
420 REM
430 IF X3<=1  THEN RS=RS+1:GOSUB 870
440 IF X3>=38 THEN LS=LS+1:GOSUB 870
450 LOCATE X3,Y3:PRINT" ";
460 COLOR 7
470 LOCATE  3,3:PRINT"LEFT SCORE=";LS;
480 LOCATE 23,3:PRINT"RIGHT SCORE=";RS;
490 IF LS=5 OR RS=5 THEN 510
500 GOTO 50
510 REM
520 LOCATE 0,7
530 PRINTSPACE$(200);
540 LOCATE 0,11
550 PRINT"              GAME OVER";
560 LOCATE 0,13
570 PRINT"          ﾓｳ1ﾄﾞ ﾔﾘﾏｽｶ?(Y/N)";
580 I$=INPUT$(1)
590 IF I$="Y" OR I$="y" THEN RUN
600 IF I$="N" OR I$="n" THEN CLS:END
610 GOTO 580
620 REM
630 CLS:COLOR 7,1
640 LOCATE 0,1:PRINT"■■■■ SHARP-X1 TV TENNIS VER 1.0 ■■■■";
650 LOCATE  3,3:PRINT"LEFT SCORE=";LS;
```

注釈:

- 120行：ボールのX, Y座標
- 140行：スペース以外はやり直し
- 110〜160行：ボールの出現する所を決める
- 180〜250行：ボールを1ヶ動かすルーチン
  - ■ 壁・ラケットなら正反射
  - ☒ ならランダムに動かす
- 280行：上へ
- 290行：下へ
- 300行：はみ出し処理
- 270〜310行：右の人のキー入力用
- 320行：右のラケットを表示
- 360行：上へ
- 370行：下へ
- 380行：はみ出し処理
- 350〜390行：左の人のキー入力用
- 400行：左のラケットを表示
- 430〜500行：どちらが勝ったの判定
- 470〜480行：スコアの表示
- 520〜610行：ゲームオーバー
- 580〜600行：再ゲームか？

7．TVテニス

```
660 LOCATE 23,3:PRINT"RIGHT SCORE=";RS;
670 PRINT
680 COLOR 3
690 PRINT"    ";STRING$(34,"▒");"    ";
700 PRINT"    ▒▒▒▒";STRING$(28," ");"▒▒▒▒    ";   壁を書く
710 LOCATE 0,21
720 PRINT"    ▒▒▒▒";STRING$(28," ");"▒▒▒▒    ";
730 PRINT"    ";STRING$(34,"▒");"    ";
740 COLOR 7
750 PRINT"  ■■ COPYRIGHT 1983.6 BY HEART SOFT ■■ ";
760 FOR I=1 TO 10
770 X=INT(RND(1)*20)+10
780 Y=INT(RND(1)*14)+7                           途中の☒を10個書く
790 LOCATE X,Y:PRINT"X";
800 NEXT I
810 RETURN                                       （画面作成ルーチン：660～810）
820 C=INT(RND(1)*2)
830 IF C=0 THEN C=-1 ELSE C=1
840 D=INT(RND(1)*2)                              ボールの進行方向を決める
850 IF D=0 THEN D=-1 ELSE D=1
860 RETURN
870 SOUND 7,28:SOUND 1,0:SOUND 3,0
880 FOR I=15 TO 0 STEP -1
890 SOUND 0,255-I*6:SOUND 2,204-I*4:SOUND 8,I:SOUND 9,I
900 SOUND 10,I/1.5:SOUND 6,30-I
910 PAUSE 1
920 NEXT I
930 RETURN
```

ゲーム終了．１対５で右側プレイヤーの勝ち！

8
陣取りゲーム

陣取りゲームは，ブロックくずしなどが出現した当時流行ったTVゲームです。当時は，CPUなどを使わずに，TTL（トランジスタ・トランジスタ・ロジック）だけで作られていたようです。その後，秋葉原のパーツ屋で〝陣取りゲームキット○○○円〞等というふうに売っているのを，見かけたことがあります。内容は，ほとんど1チップか2チップのもので，懐しさと共に半導体の進歩に新ためて驚歎したものでした。そこでX1で，これも再現してみようと思い作ってみました。陣取りゲームです。

ゲームの遊び方

あなたは，上手に自分の〝■〟赤を操って，敵の〝■〟緑の進む場所を無くせば良い訳で，自分自身・敵・壁に当たると死んでしまいます。どちらが長時間進んでいられるかを競い，先に５ポイント取った方が勝ちとなります。移動はテンキーの，２，４，６，８を使い，下，左，右，上へ方向を変えます。行番号 240の，Ｚ＝STICK（０）を，Ｚ＝STICK（１）とすれば，ジョイスティックでも楽しむことができます。このゲームの敵，Ｘ１の動きはある特徴がありますので，よく考えて移動させれば勝てるでしょう。(770 行～を見れば分かると思いますが………。)

変数表

Ｘ１，Ｙ１→自分の■のＸ，Ｙ座標

Ｙ２，Ｙ２→敵の■のＸ，Ｙ座標

ＸＸ，ＹＹ→進行方向の定義

Ｚ→キー入力（STICK文）

Ｐ→自分の進行ルーチンへ（ON　P　GOTO ～）

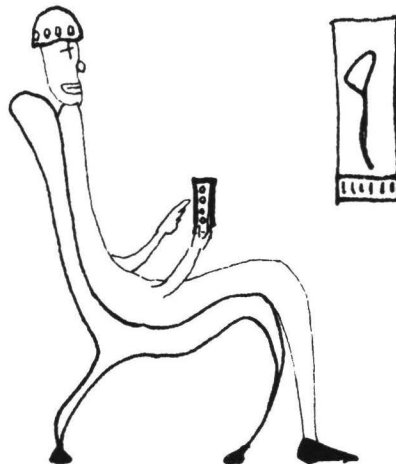
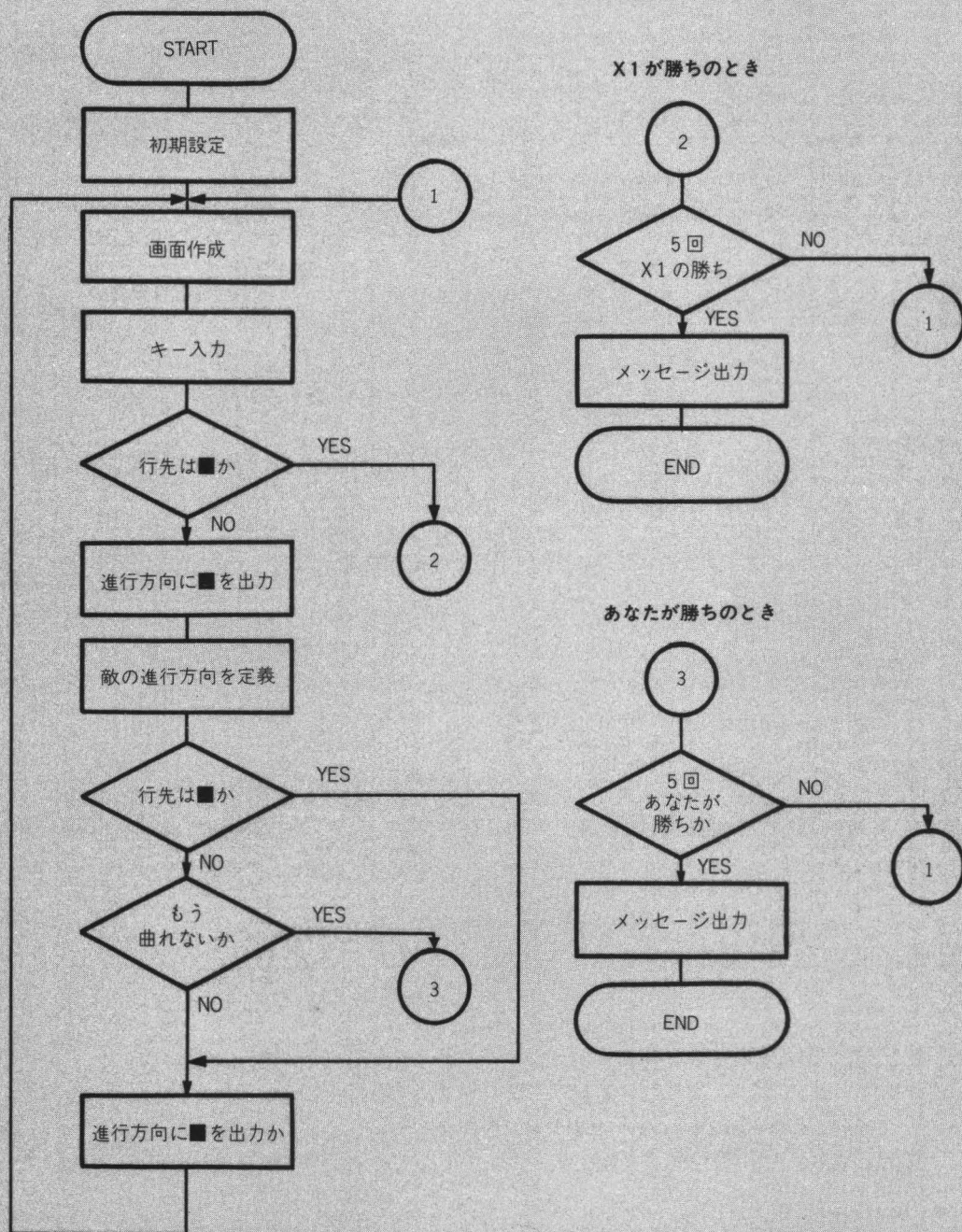
Ｒ→敵の進行ルーチンへ（ON　R　GOTO ～）

Ｉ→敵が曲がれなくなったときの判定用

ＹＳ→自分の得点

ＣＳ→敵の得点

Ｉ＄→キー入力用（INPUT＄）

陣取りゲーム・フローチャート
START
初期設定
1
画面作成
キー入力
行先は■か
YES
NO
2
進行方向に■を出力
敵の進行方向を定義
行先は■か
YES
NO
もう
曲れないか
YES
3
NO
進行方向に■を出力か
X1が勝ちのとき
2
5回
X1の勝ち
NO
1
YES
メッセージ出力
END
あなたが勝ちのとき
3
5回
あなたが
勝ちか
NO
1
YES
メッセージ出力
END

8. 陣取りゲーム

```
1 REM**************************************
2 REM*                                    *
3 REM*  ■■■■ ｼﾞﾝﾄﾘ ｹﾞｰﾑ VER 1.0 ■■■■     *
4 REM*                                    *
5 REM*      [C]Copyright 1983年 7月       *
6 REM*                                    *
7 REM*   FOR SHARP-X1 By Heart Soft       *
8 REM*                                    *
9 REM**************************************
10 CONSOLE 0,25:WIDTH 40
20 CLS:CLICK OFF:COLOR 7,0
30 RANDOMIZE:TEMPO 800
40 CLS:COLOR 7:LOCATE 0,0
50 PRINT"■■■■ SHARP-X1 ｼﾞﾝﾄﾘ ｹﾞｰﾑ VER 1.0 ■■■■";
60 LOCATE 0,24
70 PRINT"■■ [C]COPYRIGHT 1983.7 BY HEART SOFT ■■";
80 LOCATE 5,2:PRINT"YOUR SCORE=";YS
90 LOCATE 20,2:PRINT"COMPUTER SCORE=";CS
100 COLOR 5
110 LINE(0,4)-(39,22),"■",B
120 X1=INT(RND(1)*35)+2
130 Y1=INT(RND(1)*18)+5
140 IF CHARACTER$(X1,Y1)<>" " THEN 120
150 COLOR 2
160 LOCATE X1,Y1:PRINT"■";
170 X2=INT(RND(1)*35)+2
180 Y2=INT(RND(1)*18)+5
190 IF CHARACTER$(X2,Y2)<>" " THEN 170
200 COLOR 4
210 LOCATE X2,Y2:PRINT"■";
220 BEEP:PAUSE 20
230 P=INT(RND(1)*4)+1:R=INT(RND(1)*4)+1
240 Z=STICK(0)
250 IF Z=2 THEN P=1
260 IF Z=6 THEN P=2
270 IF Z=8 THEN P=3
280 IF Z=4 THEN P=4
290 ON P GOTO 370,420,470,520
300 COLOR 2
310 LOCATE X1,Y1:PRINT"■";
320 ON R GOTO 570,620,670,720
330 COLOR 4
340 LOCATE X2,Y2:PRINT"■";
350 PLAY"O4C:E"
360 GOTO 240
370 REM
380 XX=0:YY=1
390 IF CHARACTER$(X1+XX,Y1+YY)="■" THEN 1000
400 X1=X1+XX:Y1=Y1+YY
410 GOTO 300
420 REM
430 XX=1:YY=0
440 IF CHARACTER$(X1+XX,Y1+YY)="■" THEN 1000
450 X1=X1+XX:Y1=Y1+YY
460 GOTO 300
470 REM
480 XX=0:YY=-1
490 IF CHARACTER$(X1+XX,Y1+YY)="■" THEN 1000
500 X1=X1+XX:Y1=Y1+YY
510 GOTO 300
520 REM
530 XX=-1:YY=0
540 IF CHARACTER$(X1+XX,Y1+YY)="■" THEN 1000
550 X1=X1+XX:Y1=Y1+YY
560 GOTO 300
570 REM
580 XX=0:YY=1
590 IF CHARACTER$(X2+XX,Y2+YY)="■" THEN 770
600 X2=X2+XX:Y2=Y2+YY:I=0
610 GOTO 330
620 REM
630 XX=1:YY=0
640 IF CHARACTER$(X2+XX,Y2+YY)="■" THEN 770
650 X2=X2+XX:Y2=Y2+YY:I=0
```

- 10〜40：初期化ルーチン
- 110：枠を表示する
- 110〜140：自分の■を表示する
- 160〜190：もしスペースでなかったらやり直し
- 150〜210：敵の■を表示する
- 230：初めに進む方向を決める
- 240〜280：KEY入力
- STICK文を使用　ここをZ=STICK(1)にするとジョイスティックでできる
- 290：各移動ルーチンへ（自分用）
- 320：各移動ルーチンへ（敵・X1用）
- 350：移動音
- 390：壁・自分・敵に当ったときの処理
- 370〜560：自分の■の移動ルーチン
- 590：壁・自分・敵に当ったときの処理

```
660 GOTO 330
670 REM
680 XX=0:YY=-1
690 IF CHARACTER$(X2+XX,Y2+YY)="■" THEN 770
700 X2=X2+XX:Y2=Y2+YY:I=0
710 GOTO 330
720 REM
730 XX=-1:YY=0
740 IF CHARACTER$(X2+XX,Y2+YY)="■" THEN 770
750 X2=X2+XX:Y2=Y2+YY:I=0
760 GOTO 330
```

敵・X1の■の移動ルーチン

```
770 REM
780 I=I+1:IF I>4 THEN 800
790 R=I:GOTO 320
```

4回曲ろうとして曲れなければX1の負け

■に当ったときに進路変更をする

```
800 REM
810 TEMPO 800
820 PLAY"O4CDEFGABO5C"
830 YS=YS+1
840 IF YS<>5 THEN 40
```

敵X1が負けた時のルーチン

```
850 COLOR 7
860 LOCATE 5,2:PRINT"YOUR SCORE=";YS
870 LOCATE 20,2:PRINT"COMPUTER SCORE=";CS
880 COLOR 5
890 LINE(0,10)-(39,16)," ",BF
900 LOCATE 0,11
910 PRINT"                GAME OVER !!
920 LOCATE 0,13
930 PRINT"                ｱﾅﾀﾉ ｶﾁ ﾃﾞｽ｡
940 LOCATE 0,15
950 PRINT"              ﾓｳ1ﾄﾞ ﾔﾘﾏｽｶ? (Y/N)";
960 I$=INPUT$(1)
970 IF I$="Y" OR I$="y" THEN RUN
980 IF I$="N" OR I$="n" THEN CLS:END
```

あなたの勝ち

```
990 GOTO 960
1000 REM
1010 TEMPO 800
1020 PLAY"O5CO4BAGFEDC"
1030 CS=CS+1
1040 IF CS<>5 THEN 40
```

自分が負けたときのルーチン

```
1050 COLOR 7
1060 LOCATE 5,2:PRINT"YOUR SCORE=";YS
1070 LOCATE 20,2:PRINT"COMPUTER SCORE=";CS
1080 COLOR 5
1090 LINE(0,10)-(39,16)," ",BF
1100 LOCATE 0,11
1110 PRINT"                GAME OVER !!
1120 LOCATE 0,13
1130 PRINT"                X1ﾉ ｶﾁ ﾃﾞｽ｡
1140 LOCATE 0,15
1150 PRINT"              ﾓｳ1ﾄﾞ ﾔﾘﾏｽｶ? (Y/N)";
1160 I$=INPUT$(1)
1170 IF I$="Y" OR I$="y" THEN RUN
1180 IF I$="N" OR I$="n" THEN CLS:END
```

あなたの負け

```
1190 GOTO 960
```

9
砲撃ゲーム(G-RAM使用)

ある国，A国・B国があり，この 2 国は犬猿の仲で，ただ今戦争状態にあります。しかし 2 国間には高い山がそびえていて，相手側へ行くことは不可能です。その当時は科学も発達していなかったのでミサイル飛行機などはありません。あるのは単発式の大砲 1 門ずつです。この大砲は一度打つと次に打つまで時間がかかります。A国・B国順番に打ち返します。

目標は敵国の大砲です。これをやられると反撃ができなくなり戦争に負けてしまいます。うまく角度，火薬の量を調節しましょう。

このプログラムでA国の平地，B国の平地，山の高さをランダムに決めてありますので，山にならず谷になることもあります。

大砲の弾は SIN カーブを使って飛ばしています。なお判定用には，POINT 文を使用しています。

ゲームの遊び方

画面（山・平地砲台）を表示したあと角度・強さの順で入力して下さい。左右どちらから始めるかはランダムで決めています。

角度は相手側を０度，自分の真上が90度で０～90の間で入力して下さい。強さは１～30までです。なお，どちらも小数も使えます。

５ 回やって勝負を決めます。たまに弾の当った穴が変なところに出てくる場合もありますが，このプログラムでは計算上，弾のあたらないところはありません。

高得点の取り方

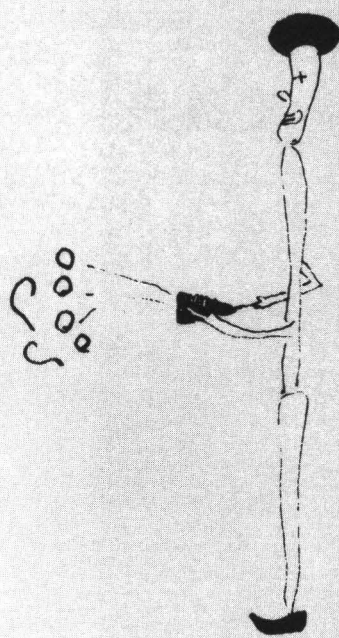
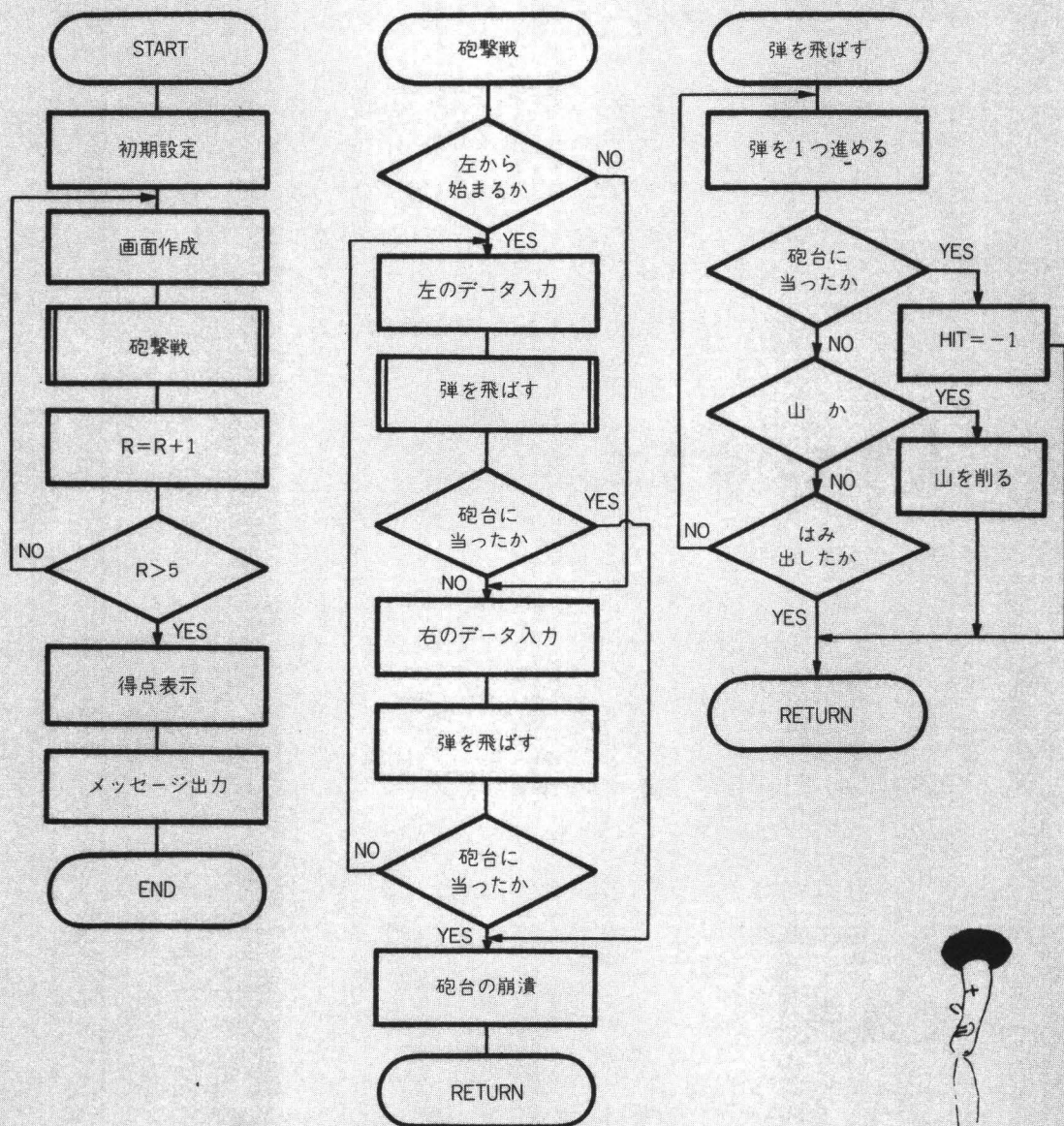
まずだいたいの見当をつけて，１ 発目を打ちます，あとは強さを調整すれば，まず当たります，あとはじゃ道なのですが，山に向けて弾を打ち，山をけずってトンネルを作る方法もあります。

何ゲームかやっているうちにだいたいの見当はついてきますョ!!

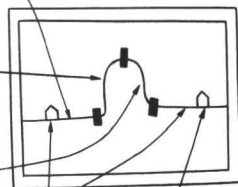
改造へのワンポイント・アドバイス

このプログラムでは，変数が沢山出てきます。ここを変化させれば，いろいろな画面が出力されるようになっています。いろいろ遊んでみましょう。

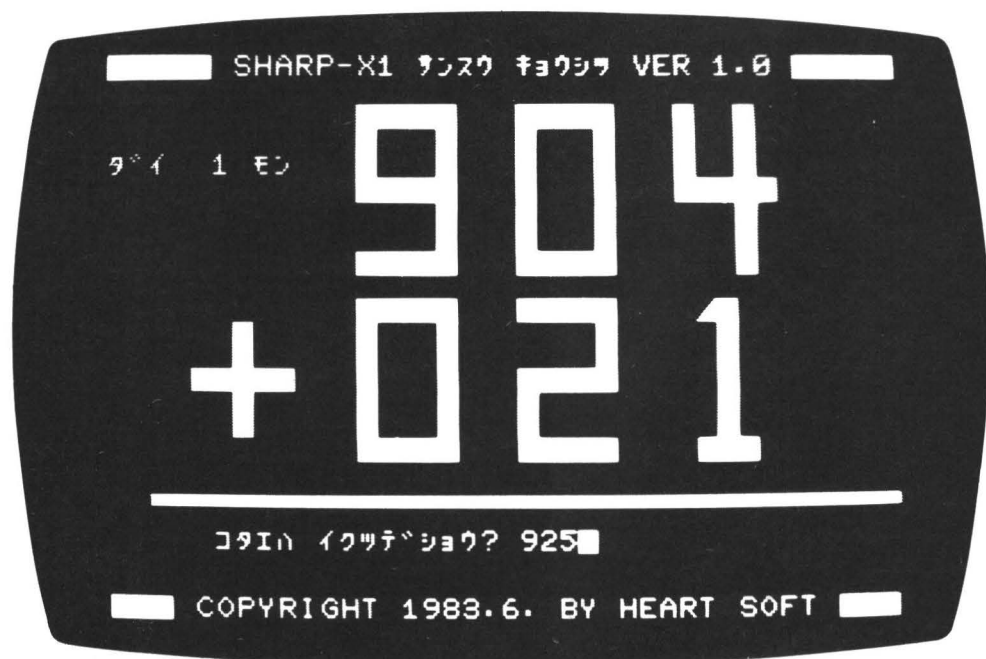
変数表

ＬＨ→左側の砲台の水平位置	ＨＨ→左右砲台のＹ座標
ＲＨ→右側　〃	Ｅ→弾の角度，当ったか否かの判定用
ＭＨ→山または谷の高さ	Ｊ＄，Ｉ＄→キー入力用
ＬＰ→左の砲台のＸ座標	Ｉ→ループ用（時間待ち）
ＲＰ→右　〃	ＤＰ→砲台の位置をずらす，仮の変数

9．砲撃ゲーム

砲撃ゲーム・フローチャート
START
初期設定
画面作成
砲撃戦
R=R+1
R>5
NO
YES
得点表示
メッセージ出力
END
砲撃戦
左から
始まるか
NO
YES
左のデータ入力
弾を飛ばす
砲台に
当ったか
YES
NO
右のデータ入力
弾を飛ばす
砲台に
当ったか
NO
YES
砲台の崩潰
RETURN
弾を飛ばす
弾を1つ進める
砲台に
当ったか
YES
HIT=−1
NO
山　か
YES
山を削る
NO
はみ
出したか
NO
YES
RETURN

```
1 REM**********************************
2 REM*                                *
3 REM*  ■■■ ﾎｳｹﾞｷ ｹﾞｰﾑ VER 1.0 ■■■   *
4 REM*                                *
5 REM*     [C]COPYRIGHT 1983年 7月     *
6 REM*                                *
7 REM*   FOR SHARP-X1 BY HEART SOFT   *
8 REM*                                *
9 REM**********************************
10 OPTION SCREEN 1:INIT"CRT:" ――― グラフィックを使えるようにする(WIDTH40で)
20 DIM A%(12)
30 GOSUB 1190 ――― PUT@用データの読み込み
40 GOSUB1080 ――― 初期データ
50 CLS4:PALET
60 FOR R=1 TO 5
70 GOSUB 150 ――― 画面作成              } メインルーチン
80 GOSUB 540 ――― 数値入力及び砲撃戦    }
90 NEXT R
100 LOCATE 3,2:COLOR 6:PRINT"LEFT SCORE=";LQ    } スコア表示
110 LOCATE22,2:COLOR 6:PRINT"RIGHT SCORE=";RQ   }
120 GOTO 1110
130 GOTO 50
140 REM
150 LH=INT(RND(1)* 80+30) ―― 左の平地の高さ      } ここを変化させれば
160 RH=INT(RND(1)* 80+30) ―― 右の平地の高さ      } 山，谷の出方が変化する
170 MH=INT(RND(1)*120+10) ―― 山または谷の高さ    }
180 IF ABS(MH-LH)<40 THEN 150  } もし山または谷と平地の差が40より小さければやり直し
190 IF ABS(MH-RH)<40 THEN 150  }
200 CLS4
210 LOCATE 3,2:COLOR 6:PRINT"LEFT SCORE=";LQ
220 LOCATE22,2:COLOR 6:PRINT"RIGHT SCORE=";RQ
230 LINE(8,199-LH)-(120,199-LH),PSET,4
240 DL=MH-LH:T=LH
250 FOR I=122 TO 160 STEP 2
260 T2=LH+((SIN((I-129-15)/15*3.1416/2)+1)/2)^2*DL
270 LINE(I-2,199-T)-(I,199-T2),PSET,4
280 T=T2
290 NEXT
300 DL=MH-RH
310 FOR I=162 TO 200 STEP 2
320 T2=RH+((-SIN((I-163-15)/15*3.1416/2)+1)/2)^2*DL
330 LINE(I-2,199-T)-(I,199-T2),PSET,4
340 T=T2
350 NEXT
360 LINE(200,199-RH)-(311,199-RH),PSET,4
370 LINE(0,0)-(319,199),PSET,4,B   } 画面の枠取りをする
380 LINE(8,8)-(311,191),PSET,4,B   }
390 PAINT(1,1),4,4                 } ぬりつぶし
400 PAINT(10,190),4,4              }
410 LP=INT(RND(1)*100)+30          } 左右の砲台の位置を決める
420 PP=LP:HH=LH:GOSUB470           }
430 RP=INT(RND(1)*100)+200         }
440 PP=RP:HH=RH:GOSUB470           }
450 RETURN
460 REM
470 LINE(PP-5,199-(HH+1))-(PP-5,199-(HH+5)),PSET,2  } 砲台を表示するルーチン
480 LINE-(PP  ,199-(HH+7)),PSET,2                   }
490 LINE-(PP+5,199-(HH+5)),PSET,2                   }
500 LINE-(PP+5,199-(HH+1)),PSET,2                   }
510 LINE-(PP-5,199-(HH+1)),PSET,2                   }
520 PAINT(PP+2,199-(HH+5)),2,2                      }
530 RETURN
540 COLOR 1
550 IF RND(1)>.5 THEN 680 ――― 左右どちらから始めるのか決める
560 LOCATE 0,22:PRINT SPACE$(39);   } 左側の角度の入力
570 LOCATE 5,22:INPUT"ｶｸﾄﾞ=";J$     } 1～90まで，それ以外はやり直し
580 IF VAL(J$)<0    THEN 560        }
590 IF VAL(J$)>=90  THEN 560        }
600 E=VAL(J$)                       }
610 LOCATE 0,22:PRINT SPACE$(39);   } 左側の強さの入力
620 LOCATE 5,22:INPUT"ﾂﾖｻ =";J$     } 1～30まで，それ以外はやり直し
630 IF VAL(J$)=0    THEN 610        }
640 IF VAL(J$)>=30  THEN 610        }
650 F=VAL(J$)                       }
```

9．砲撃ゲーム

```
660 D= 1:X=LP:Y=LH+8:GOSUB890 ———— 弾を飛ばすルーチンへ
670 IF      HIT THEN 810
680 LOCATE 0,22:PRINT SPACE$(39);        ┐
690 LOCATE 25,22:INPUT"ｶｸﾄﾞ=";J$         │
700 IF VAL(J$)=0      THEN 680            ├ 右側の角度の入力
710 IF VAL(J$)>=90  THEN 680              │
720 E=VAL(J$)                             ┘
730 LOCATE 0,22:PRINT SPACE$(39);        ┐
740 LOCATE 25,22:INPUT"ﾂﾖｻ =";J$         │
750 IF VAL(J$)=0      THEN 730            ├ 右側の強さの入力
760 IF VAL(J$)>=30  THEN 730              │
770 F=VAL(J$)                             ┘
780 D=-1:X=RP:Y=RH+8:GOSUB890 ———— 弾を飛ばすルーチンへ
790 IF NOT HIT THEN 560
800 REM
810 IF RGT= -1 THENX=LP:Y=LH:RQ=RQ+1           ┐
820 IF RGT<>-1 THENX=RP:Y=RH:LQ=LQ+1           │
830 GOSUB1240:FORI=1TO7                        │
840 PUT@(X-3,YZ-3)-(X+4,YZ+4),A%,PSET,I        ├ 弾が砲台に当ったとき
850 PUT@(X-3,YZ-3)-(X+4,YZ+4),A%,PRESET,I      │
860 NEXTI                                      │
870 RETURN                                     ┘
880 REM
890 DX=F*COS(E/180*3.1416)*D:DY=F*SIN(E/180*3.1416)*.4177 ——— 弾が飛ぶ放物線の関数
900 YZ=199-Y
910 IF POINT(X,YZ)=0                     THEN SW=FALSE  ┐ 現在飛んでいる場所の色を判定する
920 IF POINT(X,YZ)=4 OR POINT(X,YZ)=2 THEN SW=TRUE     ┘
930 PSET(X,YZ,2)          ┐
940 PRINTCHR$(7);         │
950 PSET(X,YZ,0)          ├ 弾を表示する
960 IF SW THEN 1010       ┘
970 X=X+DX:Y=Y+DY
980 IF Y<1 OR Y>199 THEN1070
990 DY=DY-.3                        ここを減らすと当り方がシビアになる
1000 GOTO900
1010 IF ABS(X-LP)<6 AND ABS(Y-LH-4)<6 THEN RGT=TRUE :HIT=TRUE:RETURN  ┐ 砲台に当る範囲
1020 IF ABS(X-RP)<6 AND ABS(Y-RH-4)<6 THEN RGT=FALSE:HIT=TRUE:RETURN ┘ ±6ドット以内
1030 IF X=>315 THENHIT=FALSE:RETURN     ┐
1040 IF X=<5   THENHIT=FALSE:RETURN     ├ はみ出し処理
1050 IF YZ=<2  THENHIT=FALSE:RETURN     ┘
1060 PUT@(X-3,YZ-3)-(X+4,YZ+4),A%,PRESET,7 ——— 山に当ったときに穴をあける
1070 HIT=FALSE:RETURN
1080 TRUE=-1:FALSE=NOT TRUE
1090 NN=72:XX=28:YY=11
1100 RETURN
1110 LOCATE 15,10:COLOR 2                 ┐
1120 PRINT" Game Over"                    │
1130 LOCATE 12,12:COLOR 1                 │
1140 PRINT"ﾓｳ1ﾄﾞ ﾔﾘﾏｽｶ?(Y/N)";            ├ ゲームオーバー
1150 I$=INPUT$(1)                         │
1160 IF I$="y" OR I$="Y" THEN RUN         │
1170 IF I$="n" OR I$="N" THEN CLS4:END    │
1180 GOTO1150                             ┘
1190 REM
1200 FOR I=0 TO 11                                                                        ┐
1210 READ A%(I):NEXT I                                                                    │
1220 DATA 15872 ,32639 ,32639 ,15999 ,15872 ,32639 ,32639 ,15999 ,15872 ,32639 ,          ├ 爆発データの読み込み
32639 ,15999                                                                              │
1230 RETURN                                                                               ┘
1240 REM ｱﾀﾘ
1250 SOUND 1,30:SOUND 0,230      ┐
1260 SOUND 3,30:SOUND 2,230      │
1270 SOUND 5,30:SOUND 4,230      │
1280 SOUND 6,31:SOUND 7,&HC8     ├ 爆発音
1290 SOUND 8,16:SOUND  9,16      │
1300 SOUND 10,16:SOUND 11,216    │
1310 SOUND 12,29:SOUND 13,0      ┘
1320 RETURN
```

10
算数教室

マイコンを使って簡単にできるＣＡＩをと思い，作ったのがこの算数教室です。〝ＣＡＩ〞とは Computer Assisted Instruction の略で，計算機を会話型式で利用した教育方式のことです。つまりコンピューターが問題を出題する先生となり，あなたが生徒になるということです。この算数教室は，小学生低学年用の加減算ですが，乗算，除算ができるようにしてみるのも良いと思います。

プログラムの使用法

問題は全部で10問，出題されます。加算，減算，数はランダムで決めていますが，常に上の数の方が，下の数より大きいように決めています。もし，これをしないと減算のとき，答えに負の数が出てしまう可能性があります。もっとも，これをしたので，下の数にゼロが出る可能性も，増えてしまいましたが……。正解の場合は画面が緑色に，不正解の場合は画面が赤色になります。全問正解しますと，最後に音楽が流れるようになっています。

変数表

I＄→キー入力用
P→ランク
M１，M２→問題
M３→＋，－の区別
X，Y→数字を出力するX，Y座標

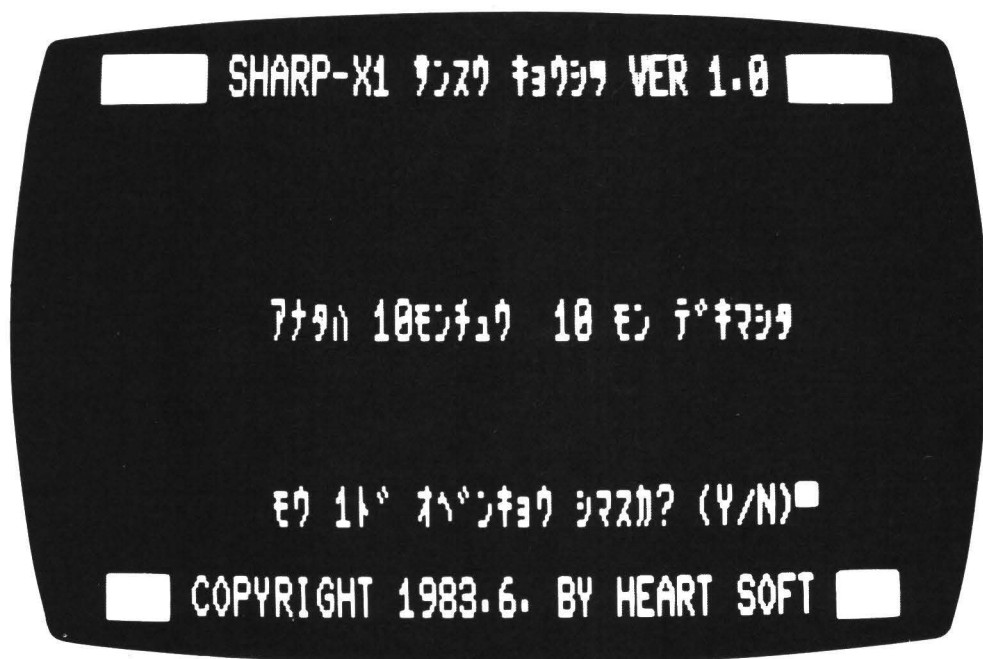
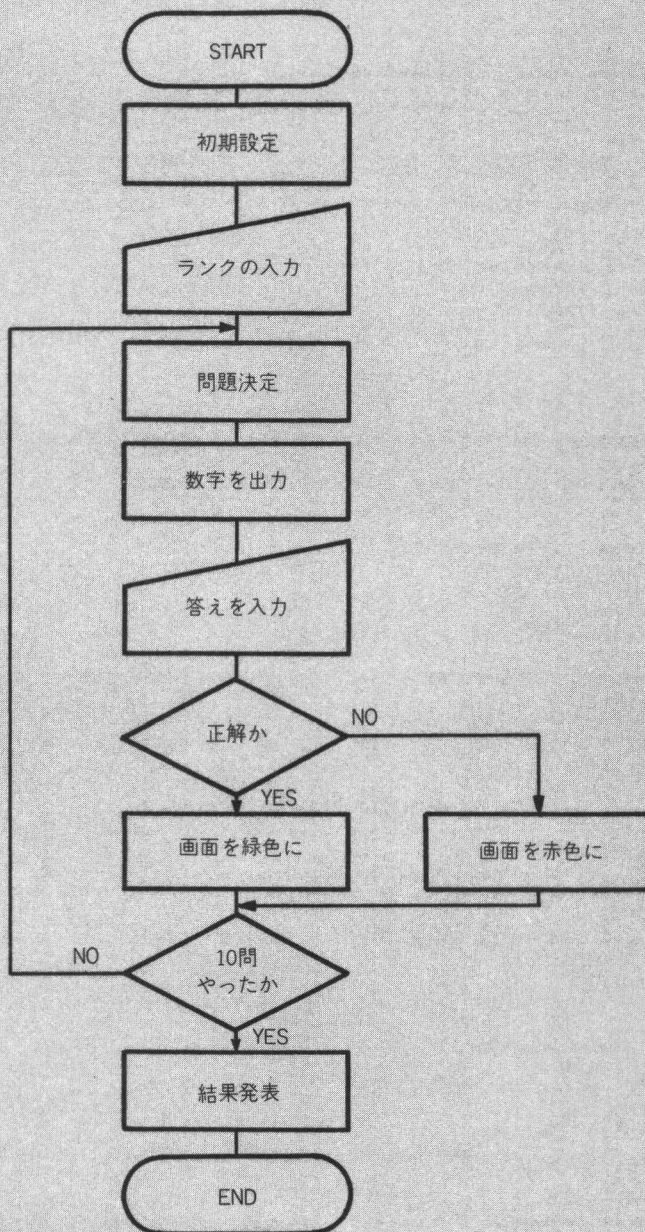
K→答えを入力する変数
Z→正解
Q→正解した数
I→ループ用

SHARP-X1 ｻﾝｽｳ ｷｮｳｼﾂ VER 1.0

ｱﾅﾀﾊ 10ﾓﾝﾁｭｳ 10 ﾓﾝ ﾃﾞｷﾏｼﾀ

ﾓｳ 1ﾄﾞ ｵﾍﾞﾝｷｮｳ ｼﾏｽｶ? (Y/N)

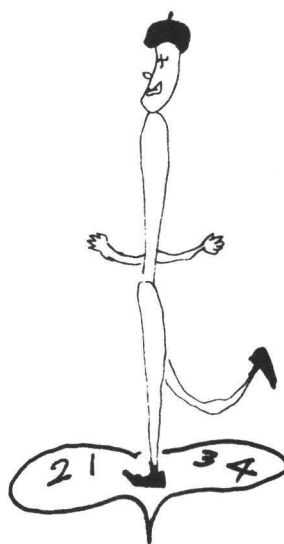
COPYRIGHT 1983.6. BY HEART SOFT

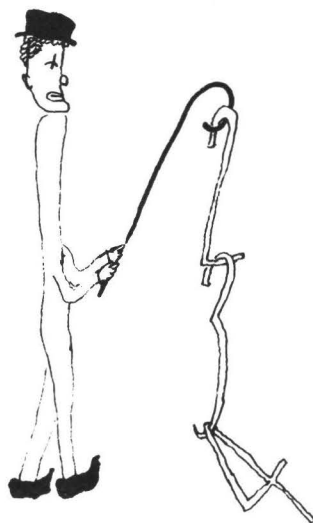
算数教室・フローチャート
START
初期設定
ランクの入力
問題決定
数字を出力
答えを入力
正解か
NO
YES
画面を緑色に
画面を赤色に
NO
10問
やったか
YES
結果発表
END

10. 算数教室

```
1 REM**************************************
2 REM*                                    *
3 REM* ■■ MATHMATIC LESSON VER 1.0 ■■ *
4 REM*                                    *
5 REM*       [C]COPYRIGHT 1983年 6月      *
6 REM*                                    *
7 REM*   FOR SHARP-X1 BY HEART SOFT       *
8 REM*                                    *
9 REM**************************************
10 WIDTH 40:CONSOLE 0,25
20 COLOR7,0:CLS:CSIZE 1
30 LOCATE 0,1
40 PRINT#0 "■■■■ SHARP-X1 ｻﾝｽｳ ｷｮｳｼﾂ、VER 1.0 ■■■■";
50 LOCATE 0,24
60 PRINT#0 "■■■ COPYRIGHT 1983.6. BY HEART SOFT ■■■";
70 CSIZE 3
80 LOCATE 12,10:PRINT#0 "1...1 ｹﾀ"
90 LOCATE 12,12:PRINT#0 "2...2 ｹﾀ"
100 LOCATE 12,14:PRINT#0 "3...3 ｹﾀ"
110 LOCATE 6,20:CSIZE 2
120 PRINT#0 "ﾄﾞﾉ ｸﾗｽﾆ ｼﾏｽｶ?";
130 I$=INPUT$(1)
140 IF I$<"1" OR I$>"3" THEN 130
150 P=VAL(I$):CLS:CSIZE 0
160 LOCATE 0,1
170 PRINT"■■■■ SHARP-X1 ｻﾝｽｳ ｷｮｳｼﾂ VER 1.0 ■■■■";
180 LOCATE 0,24
190 PRINT"■■■ COPYRIGHT 1983.6. BY HEART SOFT ■■■";
200 REM
210 FOR I=1 TO 10
220 M1=INT(RND(1)*10^P)
230 M2=INT(RND(1)*10^P)
240 IF M1<M2 THEN 220
250 M3=INT(RND(1)*2)
260 IF M3=0 THEN Z=M1+M2 ELSE Z=M1-M2
270 X=11:Y=3
280 ON (M1¥100)+1 GOSUB 1490,680,770,860,950,1040,1130,1220,1310,1400
290 M1=M1-(M1¥100)*100
300 X=19:Y=3
310 ON (M1¥10 )+1 GOSUB 1490,680,770,860,950,1040,1130,1220,1310,1400
320 M1=M1-(M1¥10 )*10
330 X=27:Y=3
340 ON  M1      +1 GOSUB 1490,680,770,860,950,1040,1130,1220,1310,1400
350 IF M3=0 THEN X=3:Y=11:GOSUB 1580 ELSE X=3:Y=11:GOSUB 1670
360 X=11:Y=11
370 ON (M2¥100)+1 GOSUB 1490,680,770,860,950,1040,1130,1220,1310,1400
380 M2=M2-(M2¥100)*100
390 X=19:Y=11
400 ON (M2¥10 )+1 GOSUB 1490,680,770,860,950,1040,1130,1220,1310,1400
410 M2=M2-(M2¥10 )*10
420 X=27:Y=11
430 ON  M2      +1 GOSUB 1490,680,770,860,950,1040,1130,1220,1310,1400
440 LOCATE 1,19
450 PRINT STRING$(38,"▂");
460 LOCATE 0,5:PRINT"ﾀﾞｲ ";I;"ﾓﾝ";
470 LOCATE 0,21:PRINT SPACE$(39);
480 LOCATE 5,21
490 INPUT"ｺﾀｴﾊ ｲｸﾂﾃﾞｼｮｳ";K
500 IF K=Z THEN COLOR ,4:Q=Q+1:GOSUB 1760 ELSE COLOR ,2:GOSUB1850
510 PAUSE 20
520 COLOR 7,0:NEXT I
530 COLOR7,0:CLS:CSIZE 1
540 LOCATE 0,0
550 PRINT#0 "■■■■ SHARP-X1 ｻﾝｽｳ ｷｮｳｼﾂ VER 1.0 ■■■■";
560 LOCATE 0,22
570 PRINT#0 "■■■ COPYRIGHT 1983.6. BY HEART SOFT ■■■";
580 LOCATE 8,10
590 PRINT#0 "ｱﾅﾀﾊ 10ﾓﾝﾁｭｳ ";Q;"ﾓﾝ ﾃﾞｷﾏｼﾀ"
600 IF Q=10 THEN GOSUB1970
610 LOCATE 8,18
620 PRINT#0 "ﾓｳ 1ﾄﾞ ｵﾍﾞﾝｷｮｳ ｼﾏｽｶ? (Y/N)";
630 I$=INPUT$(1)
640 IF I$="Y" OR I$="y" THEN RUN
650 IF I$="N" OR I$="n" THEN CLS:END
```

- 80〜140：ランクの入力1〜3まで
- 140：1〜3以外はやり直し
- 210〜260：問題を決めるルーチン
- 290：上に出る数字の百の位を表示
- 320：上に出る数字の十の位を表示
- 350：上に出る数字の一の位を表示
- 380：下に出る数字の百の位を表示
- 410：下に出る数字の十の位を表示
- 440：下に出る数字の一の位を表示
- 440〜490：答えを入力するルーチン
- 500：正解ならバックを緑色に　不正解ならバックを赤色に
- 540〜650：結果発表

```
660 GOTO 630
670 END
680 REM
690 LOCATE X,Y  :PRINT" ◢■   ";
700 LOCATE X,Y+1:PRINT"  ■   ";
710 LOCATE X,Y+2:PRINT"  ■   ";
720 LOCATE X,Y+3:PRINT"  ■   ";
730 LOCATE X,Y+4:PRINT"  ■   ";
740 LOCATE X,Y+5:PRINT"  ■   ";
750 LOCATE X,Y+6:PRINT" ■■■  ";
760 RETURN
770 REM
780 LOCATE X,Y  :PRINT"■■■■■ ";
790 LOCATE X,Y+1:PRINT"    ■ ";
800 LOCATE X,Y+2:PRINT"    ■ ";
810 LOCATE X,Y+3:PRINT"■■■■■ ";
820 LOCATE X,Y+4:PRINT"■     ";
830 LOCATE X,Y+5:PRINT"■     ";
840 LOCATE X,Y+6:PRINT"■■■■■ ";
850 RETURN
860 REM
870 LOCATE X,Y  :PRINT"■■■■■ ";
880 LOCATE X,Y+1:PRINT"    ■ ";
890 LOCATE X,Y+2:PRINT"    ■ ";
900 LOCATE X,Y+3:PRINT" ■■■■ ";
910 LOCATE X,Y+4:PRINT"    ■ ";
920 LOCATE X,Y+5:PRINT"    ■ ";
930 LOCATE X,Y+6:PRINT"■■■■■ ";
940 RETURN
950 REM
960 LOCATE X,Y  :PRINT"■ ■   ";
970 LOCATE X,Y+1:PRINT"■ ■   ";
980 LOCATE X,Y+2:PRINT"■ ■   ";
990 LOCATE X,Y+3:PRINT"■■■■■ ";
1000 LOCATE X,Y+4:PRINT"   ■  ";
1010 LOCATE X,Y+5:PRINT"   ■  ";
1020 LOCATE X,Y+6:PRINT"   ■  ";
1030 RETURN
1040 REM
1050 LOCATE X,Y  :PRINT"■■■■■ ";
1060 LOCATE X,Y+1:PRINT"■     ";
1070 LOCATE X,Y+2:PRINT"■     ";
1080 LOCATE X,Y+3:PRINT"■■■■■ ";
1090 LOCATE X,Y+4:PRINT"    ■ ";
1100 LOCATE X,Y+5:PRINT"    ■ ";
1110 LOCATE X,Y+6:PRINT"■■■■■ ";
1120 RETURN
1130 REM
1140 LOCATE X,Y  :PRINT"■■■■■ ";
1150 LOCATE X,Y+1:PRINT"■     ";
1160 LOCATE X,Y+2:PRINT"■     ";
1170 LOCATE X,Y+3:PRINT"■■■■■ ";
1180 LOCATE X,Y+4:PRINT"■   ■ ";
1190 LOCATE X,Y+5:PRINT"■   ■ ";
1200 LOCATE X,Y+6:PRINT"■■■■■ ";
1210 RETURN
1220 REM
1230 LOCATE X,Y  :PRINT"■■■■■ ";
1240 LOCATE X,Y+1:PRINT"■   ■ ";
1250 LOCATE X,Y+2:PRINT"    ■ ";
1260 LOCATE X,Y+3:PRINT"    ■ ";
1270 LOCATE X,Y+4:PRINT"    ■ ";
1280 LOCATE X,Y+5:PRINT"    ■ ";
1290 LOCATE X,Y+6:PRINT"    ■ ";
1300 RETURN
1310 REM
1320 LOCATE X,Y  :PRINT"■■■■■ ";
1330 LOCATE X,Y+1:PRINT"■   ■ ";
1340 LOCATE X,Y+2:PRINT"■   ■ ";
1350 LOCATE X,Y+3:PRINT"■■■■■ ";
1360 LOCATE X,Y+4:PRINT"■   ■ ";
1370 LOCATE X,Y+5:PRINT"■   ■ ";
1380 LOCATE X,Y+6:PRINT"■■■■■ ";
1390 RETURN
```

以下各数字及び音の出力ルーチン

10. 算数教室

```
1400 REM
1410 LOCATE X,Y  :PRINT"■■■■■ ";
1420 LOCATE X,Y+1:PRINT"■   ■ ";
1430 LOCATE X,Y+2:PRINT"■   ■ ";
1440 LOCATE X,Y+3:PRINT"■■■■■ ";
1450 LOCATE X,Y+4:PRINT"    ■ ";
1460 LOCATE X,Y+5:PRINT"    ■ ";
1470 LOCATE X,Y+6:PRINT"■■■■■ ";
1480 RETURN
1490 REM
1500 LOCATE X,Y  :PRINT"■■■■■ ";
1510 LOCATE X,Y+1:PRINT"■   ■ ";
1520 LOCATE X,Y+2:PRINT"■   ■ ";
1530 LOCATE X,Y+3:PRINT"■   ■ ";
1540 LOCATE X,Y+4:PRINT"■   ■ ";
1550 LOCATE X,Y+5:PRINT"■   ■ ";
1560 LOCATE X,Y+6:PRINT"■■■■■ ";
1570 RETURN
1580 REM
1590 LOCATE X,Y  :PRINT"      ";
1600 LOCATE X,Y+1:PRINT"  ■   ";
1610 LOCATE X,Y+2:PRINT"  ■   ";
1620 LOCATE X,Y+3:PRINT"■■■■■ ";
1630 LOCATE X,Y+4:PRINT"  ■   ";
1640 LOCATE X,Y+5:PRINT"  ■   ";
1650 LOCATE X,Y+6:PRINT"      ";
1660 RETURN
1670 REM
1680 LOCATE X,Y  :PRINT"      ";
1690 LOCATE X,Y+1:PRINT"      ";
1700 LOCATE X,Y+2:PRINT"      ";
1710 LOCATE X,Y+3:PRINT"■■■■■ ";
1720 LOCATE X,Y+4:PRINT"      ";
1730 LOCATE X,Y+5:PRINT"      ";
1740 LOCATE X,Y+6:PRINT"      ";
1750 RETURN
1760 REM ｾｲｶｲ ﾉ ｵﾄ
1770 SOUND 1,0 :SOUND 0,250
1780 SOUND 3,0 :SOUND 2,200
1790 SOUND 7,&HFC
1800 SOUND 8,16:SOUND 9,16
1810 SOUND 12,1:SOUND 11,100
1820 FOR II=0 TO 10
1830 SOUND 13,4:FOR I0=0 TO 90
1840 NEXT:NEXT:RETURN
1850 REM ﾊｽﾞﾚ ﾉ ｵﾄ
1860 SOUND 1,30 :SOUND 0,200
1870 SOUND 3,10 :SOUND 2,30
1880 SOUND 5,5  :SOUND 4,60
1890 SOUND 6,20
1900 SOUND 7,&HD8
1910 SOUND 8,12
1920 SOUND 9,16 :SOUND 10,16
1930 SOUND 12,30:SOUND 11,100
1940 SOUND 13,0:PAUSE 10
1950 SOUND 7,255:SOUND 13,0
1960 RETURN
1970 REM ｾﾞﾝﾓﾝ ｾｲｶｲ ﾉ ｵﾄ
1980 TEMPO 300:MUSIC "O5:O3V14:O6V12"
1990 MUSIC "C7DEDCO4BO5CD9r7E7D9D7r:C9C7O2B9BAAO3CC:C5rCrCrO5BrBrBrBrArArArArGrG
rGrGr"
2000 MUSIC "r7CDEDCO4B3rBrO5C7G9G:C9CO2B9AGG:O6CrCrCrCrO5BrBrA9A"
2010 RETURN
```

11
オートデータメーカー

```
60000 DATA 31,00,00,01,03,10,C5,CD
60010 DATA A3,04,3E,2A,CD,13,00,CD
60020 DATA FD,10,30,FB,1A,FE,2A,C0
60030 DATA 13,CD,50,14,13,D9,21,43
60040 DATA 10,06,0A,BE,23,28,16,23
60050 DATA 23,10,F8,D9,C9,11,73,14
60060 DATA CD,A3,04,CD,0B,00,3E,01
60070 DATA CD,EC,0D,18,C3,5E,23,56
60080 DATA D5,D9,C9,44,8B,11,4D,1D
60090 DATA 12,50,61,10,47,86,11,46
60100 DATA 53,12,52,F0,10,53,6A,10
60110 DATA 4C,9A,10,56,E1,10,54,BE
60120 DATA 12,3A,72,14,EE,01,32,72
60130 DATA 14,C9,CD,A6,12,D8,ED,43
60140 DATA 92,14,ED,53,94,14,22,96
60150 DATA 14,D5,C5,D9,CD,8B,13,21
60160 DATA 80,14,36,01,11,5A,14,CD
60170 DATA 21,13,01,20,00,CD,3B,00
60180 DATA C1,E1,D4,3E,00,3E,01,C3
60190 DATA EC,0D,CD,1F,11,E5,F5,CD
60200 DATA B5,10,F1,E1,30,03,2A,14
60210 DATA FF,ED,4B,12,FF,CD,44,00
```

このプログラムは，X1のRAM上のデータを読み出して，BASICのDATA文に直すプログラムです。機械語の，ちょっとしたサブルーチンなどを，BASICのプログラムといっしょに，まとめたいときなどに便利だと思います。逆にこのプログラムを使ってできた，DATA文をまた元へ戻す方法も紹介しますので参考にして下さい。なお，このプログラムは自分自身で，自分を書き加える機能を持っている自己増殖型のプログラムです。これには，KEY 0,―を利用しています。みなさんがプログラムを作るときの参考になると思いますので活用してみて下さい。

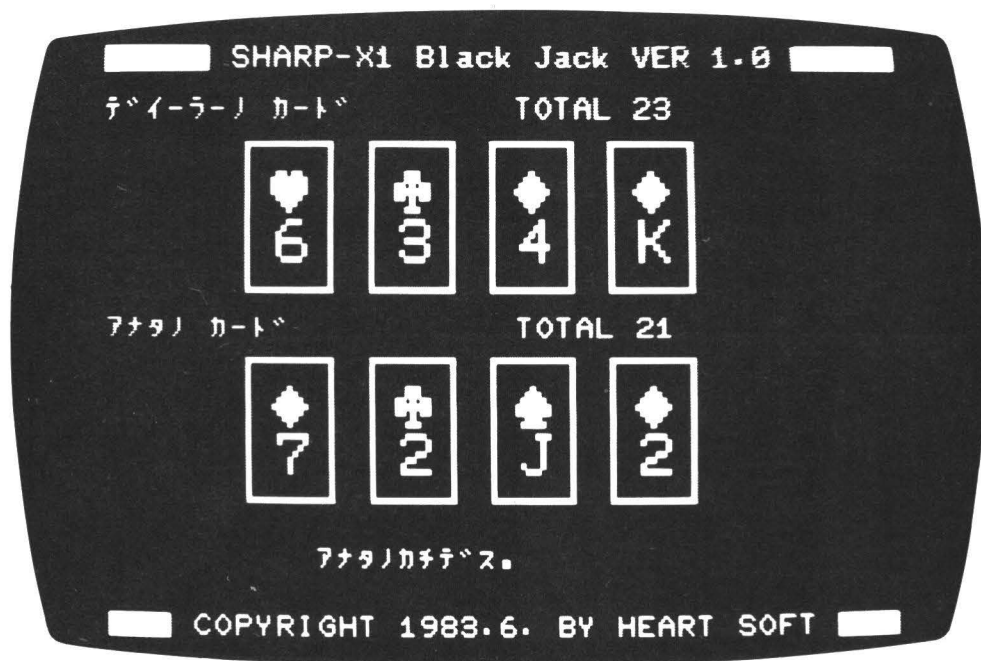
プログラムの使用法

DATA 文にしたい先頭のアドレスと，最後のアドレスを16進で打ち込んで下さい。すると，自動的に画面に DATA が表示されて読み込んで行くのが見えると思います。DATA 文を作り終ると，OKが出ますので，LIST を取ってみて下さい。60000 行から DATA 文が入っていると思います。なお，この DATA 文を元に戻す方法ですが，たとえば &H1000～&H1FFF までを元にもどすとすれば，

```
10  FOR   I=&H1000  TO  &H1FFF
20  READ  I$
30  POKE  I,VAL("&H"+I$)
40  NEXT  I
```

上記のようにして下さい。

```
1 REM*************************************
2 REM*                                   *
3 REM* ■ AUTO DATA MAKER VER 1.0 ■      *
4 REM*                                   *
5 REM*      [C]COPYRIGHT 1983年 7月      *
6 REM*                                   *
7 REM*  FOR SHARP-X1 BY HEART SOFT       *
8 REM*                                   *
9 REM*************************************
10 CONSOLE 0,25:WIDTH 40
20 COLOR 7:CLS
30 INPUT"START ADDRESS";ST$          ┐
40 INPUT"END   ADDRESS";EN$          │ 数値の入力
50 ST=VAL("&H"+ST$)                  │
60 EN=VAL("&H"+EN$)                  ┘
70 LN=60000!  ─────────────── 先頭の行番号
80 CLS:P=0
90 PRINT:PRINT LN;"DATA ";                         ┐
100 PRINT RIGHT$("00"+HEX$(PEEK(ST+P)),2);         │ 60000 DATA —, —, —, ……—, —
110 PRINT",";                                      │     というものを画面に表示する
120 P=P+1:IF P<>7 THEN 100                         │
130 PRINT RIGHT$("00"+HEX$(PEEK(ST+P)),2);         ┘
140 LN=LN+10  ─────────────── 行番号を増やす
150 IF ST>=EN THEN END
160 ST=ST+8
170 PRINT
180 PRINT"LN=";LN;"ST=";ST;"EN=";EN ─── 変数を保護する
190 PRINT:PRINT"GOTO 80"  ─────────── 再スタートする行番号
200 PRINT CHR$(11);
210 KEY0,CHR$(13)+CHR$(13)+CHR$(13) ─── X1にRETURNを3回押させる
```

12
BLACK JACK

トランプを使ったギャンブル・ゲームの中ではポーカーと並んで最もポピュラーなゲームです。

ラスベガスでは，このゲームで負け路頭に迷う人も数多いそうですがここではいくら負けても何も取られません。どうぞ安心してお遊び下さい。

ゲームの遊び方

このゲームの勝敗はカードの合計が 21 に近い方が勝ちというゲームです。21 をオーバーすると無条件で負けてしまいます。絵札は 10 と数えてエースは 1 または 11 と自分の都合の良い方に数えることができます。カードを 1 枚見たところで，その回のかけ金を決めます。2 枚目のカードを見た後 5 枚まで追加のカードを，もらうことができます。親は 16 以上になるまで，カードをひきます。あなたのカードの合計と親のカードの合計が同じ場合は引き分けとなります。10,000円以上もうけると，親が破産してしまいます。もち金が 0 になればあなたの破産です。

高得点の取り方

このプログラムでは 1 枚目のカードを見たところで，かけ金を決められるので，1 枚目が絵札などの有利なカードのときは沢山のかけ金を逆に，2～3 などの不利なカードのときは控目にかけましょう。

また親のカードが A などの場合もそうです。親は 16 未満だと必ず引いてくるので，21 をオーバーしそうなときでもチャンスです。

以上のことを守り慎重に賭ければ必ず勝てるでしょう。

改造へのワンポイント・アドバイス

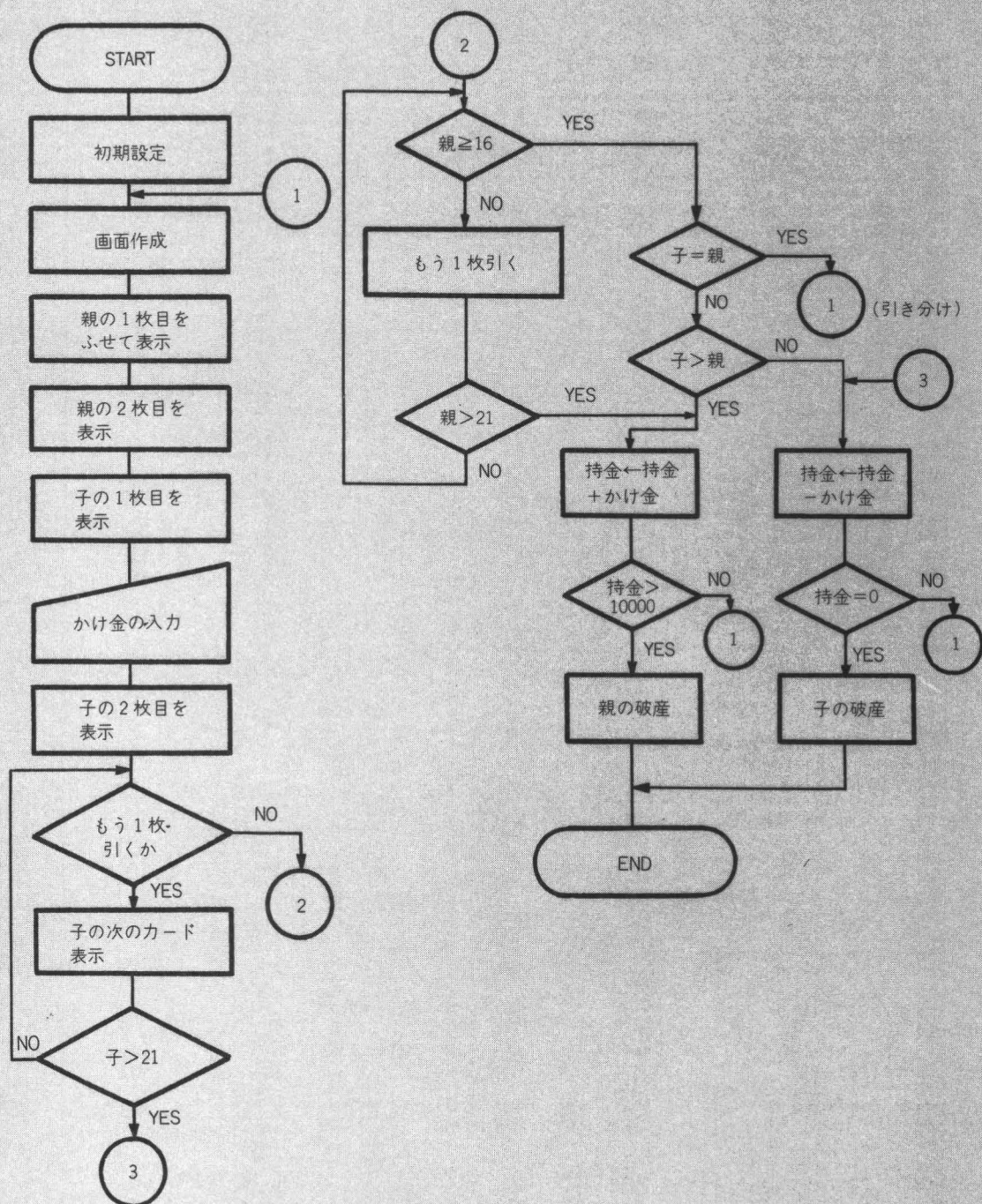
親・子共 5 枚までしか札が表示されません。これも画面上の問題なのですが，札を重ね合わせるなどして，もっと多く札を出す方法もあります。

変　数

X(n)→親のカードの数
Z(n)→親のカードのマーク
Y(n)→3 のカードの数
W(n)→3 のカードのマーク
F(n, n)→同じカードが出ないための配列
A$, B$→カードプリントルーチンでのマーク，数値

T1→親のトータル
T2→子のトータル
I→ループ用（時間待ち）

L→カードの枚数
Q→持ち金

ブラックジャック・フローチャート
START
初期設定
1
画面作成
親の1枚目を
ふせて表示
親の2枚目を
表示
子の1枚目を
表示
かけ金の入力
子の2枚目を
表示
もう1枚
引くか
NO
2
YES
子の次のカード
表示
NO
子>21
YES
3
2
親≧16
YES
NO
もう1枚引く
親>21
YES
NO
子=親
YES
1
(引き分け)
NO
子>親
NO
3
YES
持金←持金
+かけ金
持金←持金
ーかけ金
持金>
10000
NO
1
YES
親の破産
持金=0
NO
1
YES
子の破産
END

12. BLACK JACK

```
1 REM************************************
2 REM*                                  *
3 REM*   ■■■■ BLACK JACK VER 1.0 ■■■■   *
4 REM*                                  *
5 REM*      [C] COPYRIGHT 1983年 7月     *
6 REM*                                  *
7 REM*   FOR SHARP-X1 BY HEART SOFT     *
8 REM*                                  *
9 REM************************************
10 DIM X(5),Y(5),Z(5),W(5),F(5,13)
20 Q=100                                  ┐
30 FOR I=1 TO 4:FOR J=1 TO 13            ├ 変数の初期化をする
40 F(I,J)=0:NEXT J:NEXT I                │
50 T1=0:E=0                               ┘
60 CLS:LOCATE 0,0:COLOR7:PRINT"■■■■ SHARP-X1 Black Jack VER 1.0 ■■■■";  ┐ タイトル表示
70 LOCATE 0,24:PRINT"■■■ COPYRIGHT 1983.6. BY HEART SOFT ■■■"          ┘
80 LOCATE 0,2:COLOR 6:PRINT"ディーラーノ カード";
90 A$=" ":B$=" ":C=6
100 V=3:GOSUB 1380         ――― 親の1枚目をふせて表示する
110 FOR I=1 TO 5                            ┐
120 Z(I)=INT(RND(1)*4)+1:J=Z(I) ― マーク    │
130 X(I)=INT(RND(1)*13)+1:K=X(I) ― 数字     ├ 親のカードを5枚分決めておく
140 IF F(J,K)=-1 THEN 150                   │
150 F(J,K)=-1:NEXT I                        ┘
160 IF X(2)=1 THEN GOSUB 1070 ――― Aの処理
170 T1=T1+X(2):A=2
180 V=3:GOSUB 1150             ――― 親のカードを表示する
190 T2=0:D=0
200 LOCATE 0,11:COLOR 6:PRINT"アナタノ カード"
210 A$=" ":B$=" ":C=6:V=12:GOSUB 1380 ――― 子の1枚目をふせて表示する
220 FOR I=1 TO 5                            ┐
230 Y(I)=INT(RND(1)*4)+1:J=Y(I) ― マーク    │
240 W(I)=INT(RND(1)*13)+1:K=W(I) ― 数字     ├ 子のカードを5枚分決めておく
250 IF F(J,K)=-1 THEN 230                   │
260 F(J,K)=-1:NEXT I                        ┘
270 IF W(2)=1 THEN GOSUB 1110 ――― Aの処理
280 T2=T2+W(2):B=2
290 V=12:GOSUB 1200            ――― 子のカードを表示する
300 COLOR 7
310 LOCATE 0,20:PRINT SPACE$(39);           ┐
320 LOCATE 5,20:PRINT"$";Q;"モッテイマス"   │
330 PAUSE 10                                │
340 LOCATE 20,20:PRINT SPACE$(20);          ├ 賭金の入力
350 LOCATE 20,20:INPUT"イクラカケマスカ ";M │
360 M=INT(M)                                │
370 IF (Q<M)+(M<1) THEN 340 ――――――――――――――┘― 持金より大きかったり，1より小さければやり直し
380 IF W(1)=1 THEN GOSUB 1110 ――― Aの処理
390 T2=T2+W(1):B=1
400 V=12:GOSUB 1200
410 B=2
420 LOCATE 0,20:PRINT SPACE$(39);
430 LOCATE 10,21:COLOR 7:PRINT"モウ1マイヒキマスカ(Y/N)";  ┐
440 C$=INPUT$(1)                                            ├ もう1枚引くか？
450 IF C$="n" OR C$="N" THEN 610                            │ キー入力
460 IF C$="y" OR C$="Y" THEN 480                            ┘
470 GOTO 420
480 B=B+1:D=0
490 IF B>5 THEN B=5:GOTO 610
500 IF W(B)=1 THEN GOSUB 1110 ――― Aが入っていたときの処理
510 T2=T2+W(B)
520 IF T2<=21 THEN 590
530 FOR I=1 TO B
540 IF W(I)=1 THEN W(I)=0:T2=T2-10:GOTO 590
550 NEXT I
560 GOSUB 1200
570 IF T2<22 THEN 430
580 GOTO 910
590 GOSUB 1200
600 GOTO 430
610 IF X(1)=1 THEN GOSUB 1070
620 T1=T1+X(1):A=1
630 V=3:GOSUB 1150
640 A=2
```

```
650 IF T1<16 THEN 700  ――――――― 親のカードが16未満のときはもう１枚
660 IF T1<T2 THEN 850  ――――――― あなたの勝ち
670 IF T1>T2 THEN 910  ――――――― ディーラーの勝ち
680 LOCATE 0,21:PRINT SPACE$(39);
690 LOCATE 10,21:COLOR 3:PRINT"ﾋｷﾜｹﾃﾞｽ｡":PAUSE 20:GOTO 940  } 引き分け
700 PAUSE 10
710 A=A+1:E=0
720 IF A>5 THEN 830
730 IF X(A)=1 THEN GOSUB 1070
740 T1=T1+X(A):IF T1<22 THEN 810
750 FOR I=1 TO A
760 IF X(I)=1 THEN X(I)=0:T1=T1-10:GOTO 810
770 NEXT I
780 GOSUB 1150
790 IF T1<22 THEN 650
800 GOTO 850
810 GOSUB 1150
820 GOTO 650
830 IF B=6 THEN Q=Q+2*M:GOTO 910
840 IF B=5 THEN Q=Q+M:GOTO 910
850 IF T2=21 THEN Q=Q+M
860 LOCATE 0,21:PRINT SPACE$(39);
870 Q=Q+M:LOCATE10,21:COLOR 4:PRINT"ｱﾅﾀﾉｶﾁﾃﾞｽ｡"   } あなたの勝ち
880 TEMPO 1250
890 MUSIC"L1505GBFEGBFEGBFEGG:E04A05DCE04A05DCE04A05DCDD"
900 GOTO 940
910 LOCATE 0,21:PRINT SPACE$(39);
920 Q=Q-M:LOCATE10,21:COLOR 2:PRINT"ﾃﾞｨｰﾗｰﾉｶﾁﾃﾞｽ｡"  } ディーラーの勝ち
930 MUSIC "L1504FEFDEDC"
940 IF Q<1 THEN 990
950 IF Q<10000 THEN 30
960 LOCATE 10,20:COLOR 2:PRINT"ﾃﾞｨｰﾗｰﾊ ﾊｻﾝﾃﾞｽ｡"
970 LOCATE 10,21:COLOR 7:PRINT"   Give Up!!    "   } ディーラーの破産
980 GOTO 1020
990 REM
1000 LOCATE 10,20:COLOR 2:PRINT"   ｱﾅﾀﾊ ﾊｻﾝﾃﾞｽ｡"
1010 LOCATE 10,21:COLOR 7:PRINT"   Game Over    "  } あなたの破産
1020 LOCATE 10,22:COLOR 7:PRINT"ﾓｳ1ﾄﾞﾔﾘﾏｽｶ(Y/N)";
1030 I$=INPUT$(1)
1040 IF I$="y" OR I$="Y" THEN RUN                  } キー入力
1050 IF I$="n" OR I$="N" THEN CLS:END
1060 GOTO 1030
1070 REM
1080 T1=T1+10
1090 IF T1>20 THEN T1=T1-10:E=-1
1100 RETURN
1110 REM                               } Aの処理ルーチン
1120 T2=T2+10
1130 IF T2>20 THEN T2=T2-10:D=-1
1140 RETURN
1150 REM
1160 X1=X(A):Z1=Z(A)
1170 IF E=-1 THEN X(A)=0               } 親のカードを表示するルーチン
1180 IF X1>10 THEN T1=T1+10-X1
1190 C=6*A     :T3=T1:GOTO 1250
1200 REM
1210 X1=W(B):Z1=Y(B)
1220 IF D=-1 THEN W(B)=0               } 子のカードを表示するルーチン
1230 IF X1>10 THEN T2=T2+10-X1
1240 C=6*B     :T3=T2
1250 REM
1260 IF X1=1 THEN B$="A":GOTO 1320
1270 IF X1=10 THEN B$="X":GOTO 1320
1280 IF X1=11 THEN B$="J":GOTO 1320      } 絵札用判定ルーチン
1290 IF X1=12 THEN B$="Q":GOTO 1320
1300 IF X1=13 THEN B$="K":GOTO 1320
1310 A$="0":B$=CHR$(ASC(A$)+X1)
1320 IF Z1=1 THEN A$="♠":H=1:GOTO 1360
1330 IF Z1=2 THEN A$="♥":H=2:GOTO 1360   } マークを決めるルーチン
1340 IF Z1=3 THEN A$="♦":H=2:GOTO 1360
1350 A$="♣":H=1
1360 LOCATE 20,V-1:COLOR 5:PRINT"TOTAL";T3
1370 COLOR H:PRINT CHR$(7);
```

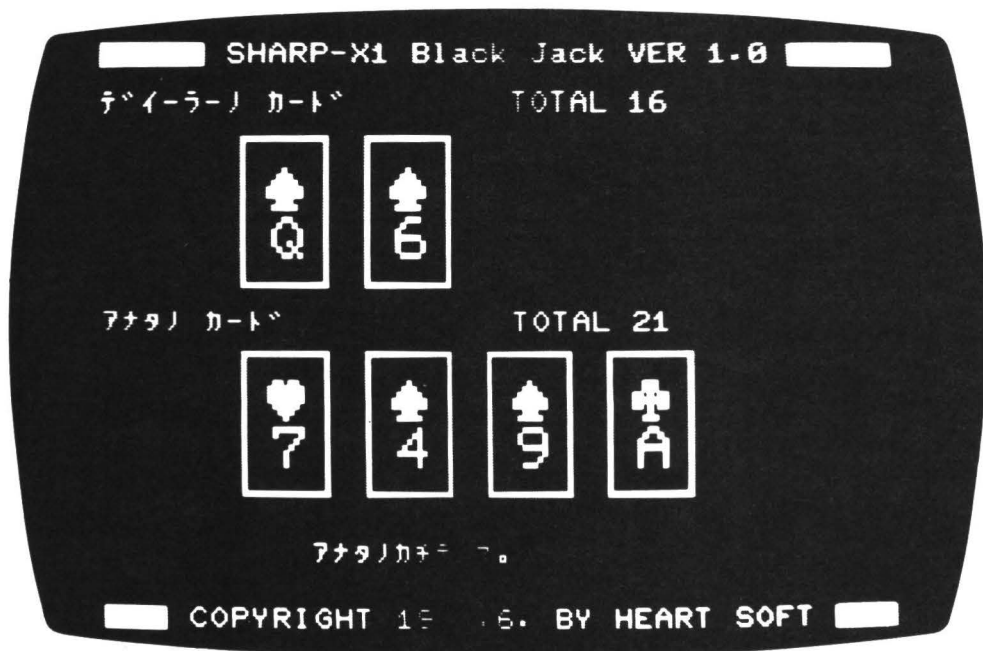
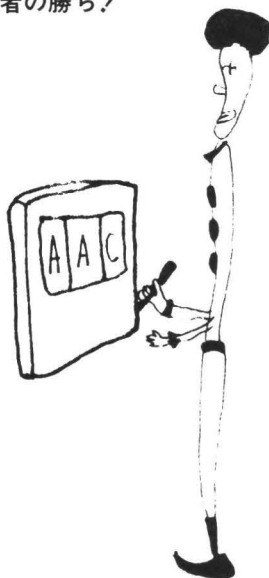
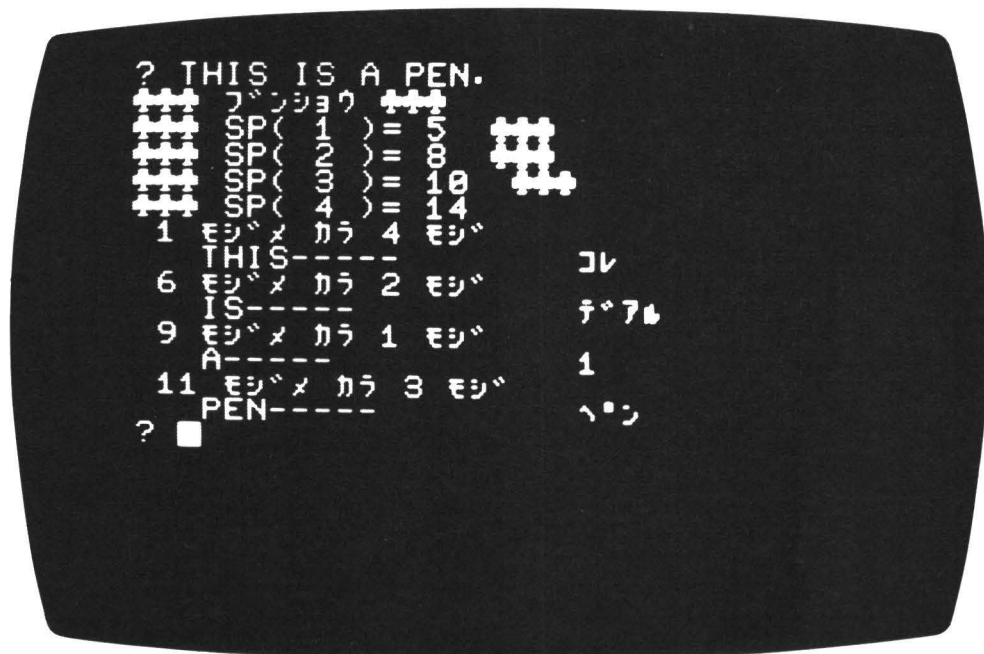
12. BLACK JACK

```
1380 REM
1390 CSIZE 3
1400 LOCATE C,V  :PRINT #0,"┌─┐"
1410 LOCATE C,V+2:PRINT #0,"|"A$"|"
1420 LOCATE C,V+4:PRINT #0,"|"B$"|"
1430 LOCATE C,V+6:PRINT #0,"└─┘"
1440 CSIZE 0
1450 RETURN
```

（1400〜1430）カード表示用サブルーチン

挑戦者の勝ち!

13
自動簡易翻訳機

```
? THIS IS A PEN.
### ブンショウ ###
### SP( 1 )= 5
### SP( 2 )= 8
### SP( 3 )= 10
### SP( 4 )= 14
 1 モジメ カラ 4 モジ
   THIS-----              コレ
 6 モジメ カラ 2 モジ
   IS-----                デアル
 9 モジメ カラ 1 モジ
   A-----                 1
 11 モジメ カラ 3 モジ
   PEN-----               ペン
? ■
```

数年前 **SHARP** から電子翻訳機が登場したときは非常に驚いてしまいました。何と言っても電卓ほどの大きさのものに辞書が入ってしまったのですから，それ以来文章の訳せるもの海外旅行用のものなども登場し，つい最近になると，とうとう“ウォーキング・ディクショナリー”などといって，時計の中へ入ってしまいました。わがＸ１もコンピューターの端くれ,“電卓もどきなどや，時計にできてコンピューターにできないことはない”と意気込んで作ったのがこの電訳機もどきです。

しかし作者の力不足でまともに訳してくれません。たとえば「This is a pen.」を入力すると「これ,です，１つ，ペン」と訳してしまいます。これは名詞，動詞などの区別をしていないためで，のちにバージョンアップすることができるようにＣ＄(ｎ)に判定用のデータを入れておきました。自信のある人はためしてみて下さい。

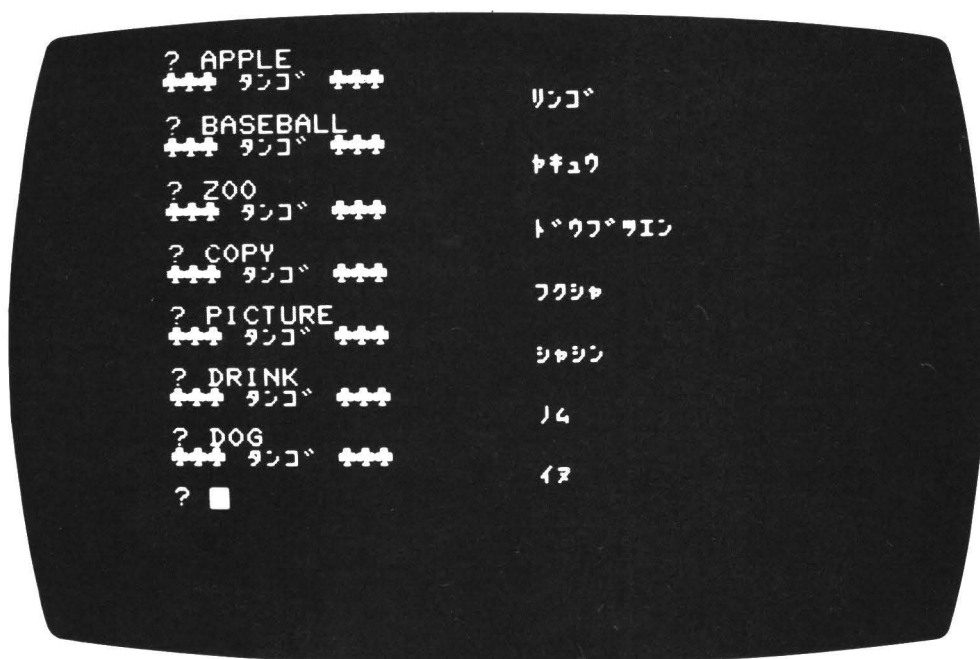
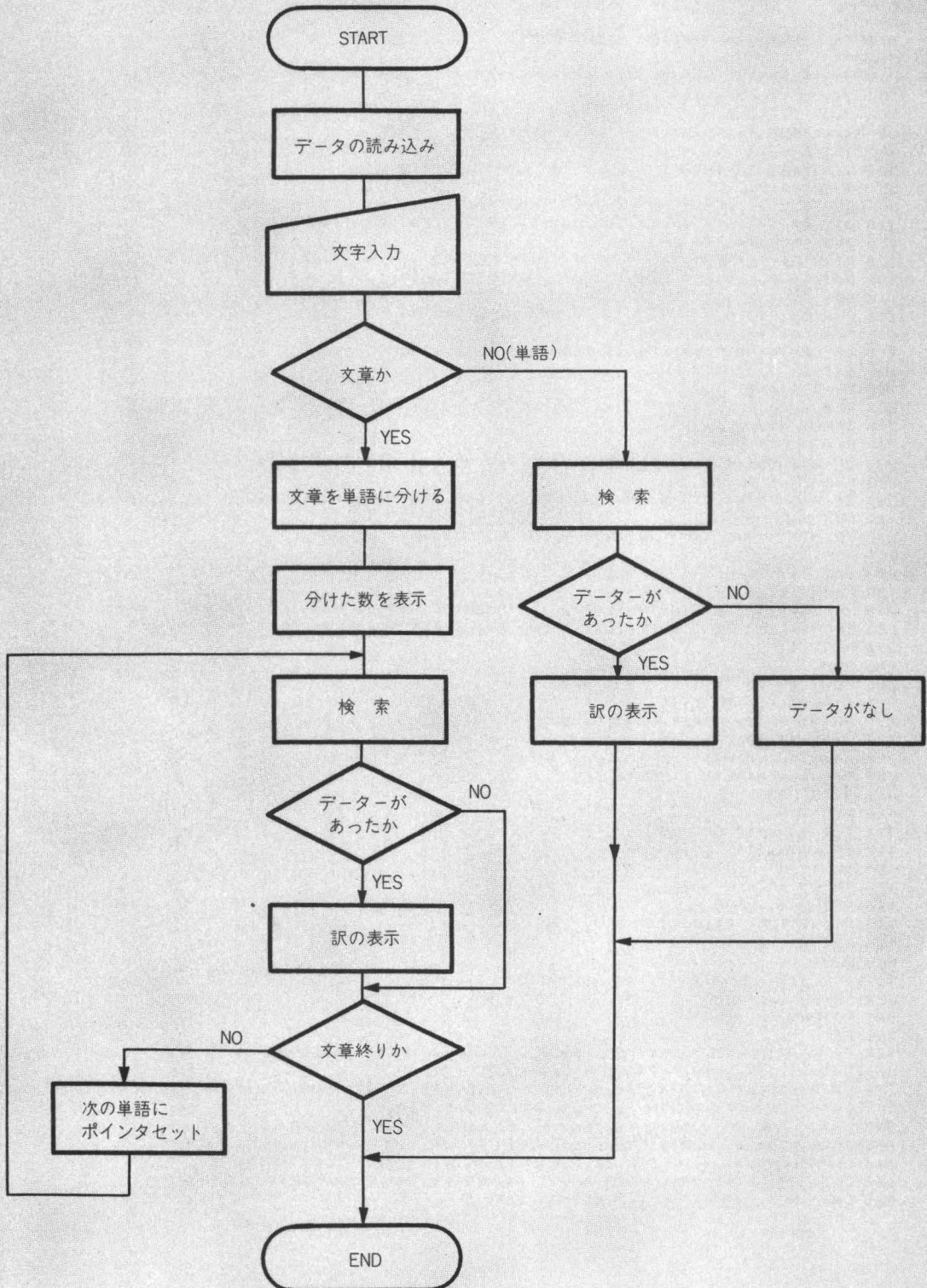
単語を増したい人は（ｎ）の値を増やして下さい。これは単語の数です。

プログラムの使用方法

単語を訳すときはそのまま，文章を訳すときは最後に必ず．（ピリオド）を入力して下さい。そうでないと文全体が単語として出てしまいます。なお，文章の初めを大文字にする必要はありません。

変数表

A $(n) →単語
B $(n) →訳
C $(n) →判定用
W $ →訳した文字例
P →単語データの数
F $(n) →アルファベットの順
L →入力した文字例の長さ
X $ →翻訳する文字列
SP(n) →文章を単語にくぎるときに使用
I →FOR 文で使用

自動簡易翻訳機・フローチャート
START
データの読み込み
文字入力
文章か
NO(単語)
YES
文章を単語に分ける
検　索
分けた数を表示
データーが
あったか
NO
YES
訳の表示
データがなし
検　索
データーが
あったか
NO
YES
訳の表示
NO
文章終りか
次の単語に
ポインタセット
YES
END

13. 自動簡易翻訳機

```
1 REM************************************
2 REM*                                  *
3 REM* ■■■ ﾃﾞﾝｼ ｼﾞｼｮ ﾓﾄﾞｷ VER 1.0 ■■■ *
4 REM*                                  *
5 REM*      [C] COPYRIGHT 1983年 7月    *
6 REM*                                  *
7 REM*   FOR SHARP-X1 BY HEART SOFT     *
8 REM*                                  *
9 REM************************************
10 CONSOLE 0,25:WIDTH 40
20 COLOR 7,0:CLS
30 PRINT"■■■ SHARP-X1 ﾃﾞﾝｼ ｼﾞｼｮ ﾓﾄﾞｷ VER 1.0 ■■■";
40 LOCATE 0,23
50 PRINT"■■■ COPYRIGHT 1983.7. BY HEART SOFT ■■■";
55 PAUSE 10:CLS
60 LOCATE 0,3
100 DIM A$(50),B$(50),C$(50),D$(50),E$(50),F$(50),G(50),SP(50)
110 Q=39:FOR I=1 TO 26
120 READ F$(I):NEXT I
130 DATA A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P,Q,R,S,T,U,V,W,X,Y,Z
140 REM
150 FOR I=1 TO Q
160 READ A$(I),B$(I),C$(I)
170 IF LEFT$(A$(I),1)=F$(J) THEN G(J)=G(J)+1:GOTO 190
180 J=J+1
190 NEXT I:J=0
200 REM
210 INPUT X$:L=LEN(X$)
220 REM
230 IF RIGHT$(X$,1)="." THEN PRINT"♣♣♣ ﾌﾞﾝｼｮｳ ♣♣♣":GOTO 280
240 FOR JF=1 TO 26
250 IF RIGHT$(X$,1)=F$(JF) THEN PRINT"♣♣♣ ﾀﾝｺﾞ ♣♣♣":W$=X$:GOSUB 420:GOTO 280
260 NEXT JF
270 PRINT"ﾆｭｳﾘｮｸ ﾐｽ ﾃﾞｽ｡":GOSUB 500:GOTO 200
280 REM
290 FOR I=1 TO L
300 M$=MID$(X$,I,1)
310 IF M$=" " THEN IS=IS+1:SP(IS)=I:BEEP:PRINT"♣♣♣ SP("IS")="SP(IS)" ♣♣♣"
320 IF M$="." THEN IS=IS+1:SP(IS)=I:PRINT"♣♣♣ SP("IS")="SP(IS):GOTO 340
330 NEXT I
340 REM
350 FOR I=0 TO IS-1
360 S1=SP(I+1)-SP(I)-1
370 PRINT SP(I)+1"ﾓｼﾞﾒ ｶﾗ";S1"ﾓｼﾞ"
380 W$=MID$(X$,SP(I)+1,S1)
390 PRINT"   ";W$;"-----";
400 GOSUB 420:NEXT I:GOSUB 500
410 GOTO 200
420 REM
430 FOR IA=1 TO Q
440 IF W$=A$(IA) THEN 480
450 NEXT IA
460 PRINT"ﾃﾞｰﾀｶﾞ ｱﾘﾏｾﾝ｡"
470 RETURN
480 PRINT TAB(20);B$(IA)
490 RETURN
500 REM
510 FOR II=1 TO Q:SP(II)=0:NEXT II
520 IS=0
530 RETURN
540 REM
550 DATA A,1,N,AM,ﾃﾞｱﾙ,V,BOY,ｼｮｳﾈﾝ,N,DOG,ｲﾇ,N,CITY,ﾏﾁ,N,HE,ｶﾚ,N
560 DATA I,ﾜﾀｼ,N,IS,ﾃﾞｱﾙ,V,THIS,ｺﾚ,N,YOU,ｱﾅﾀ,N
570 DATA AGO,ﾑｶｼ,N,AIR,ｸｳｷ,N,ALONG,ｿｯﾃ,V,ARE,ﾃﾞｱﾙ,V,WAS,ﾃﾞｱｯﾀ,V
580 DATA VERY,ﾄﾃﾓ,N,KIND,ｼﾝｾﾂ,N,UNDERSTAND,ﾘｶｲ,N
590 DATA HAVE,ﾓﾂ,V,DO,ｽﾙ,V,PLAY,ｱｿﾌﾞ,V,BALL,ﾎﾞｰﾙ,N,BASEBALL,ﾔｷｭｳ,N
600 DATA BOOK,ﾎﾝ,N,PICTURE,ｼｬｼﾝ,N,PEN,ﾍﾟﾝ,N,WINDOW,ﾏﾄﾞ,N,KICK,ｹﾙ,V
610 DATA COPY,ﾌｸｼｬ,V,LOOK,ﾐﾙ,V,WANT,ﾎｯｽ,V,CRY,ｻｹﾌﾞ,V,DRINK,ﾉﾑ,V
620 DATA ZOO,ﾄﾞｳﾌﾞﾂｴﾝ,N,APPLE,ﾘﾝｺﾞ,N,WERE,ﾃﾞｱｯﾀ,V,ORANGE,ﾐｶﾝ,N
630 DATA PET,ｱｲｶﾞﾝﾄﾞｳﾌﾞﾂ,N,COSMOS,ﾖｺﾊﾏ,N
```

100〜130: アルファベットのデータを読み込む

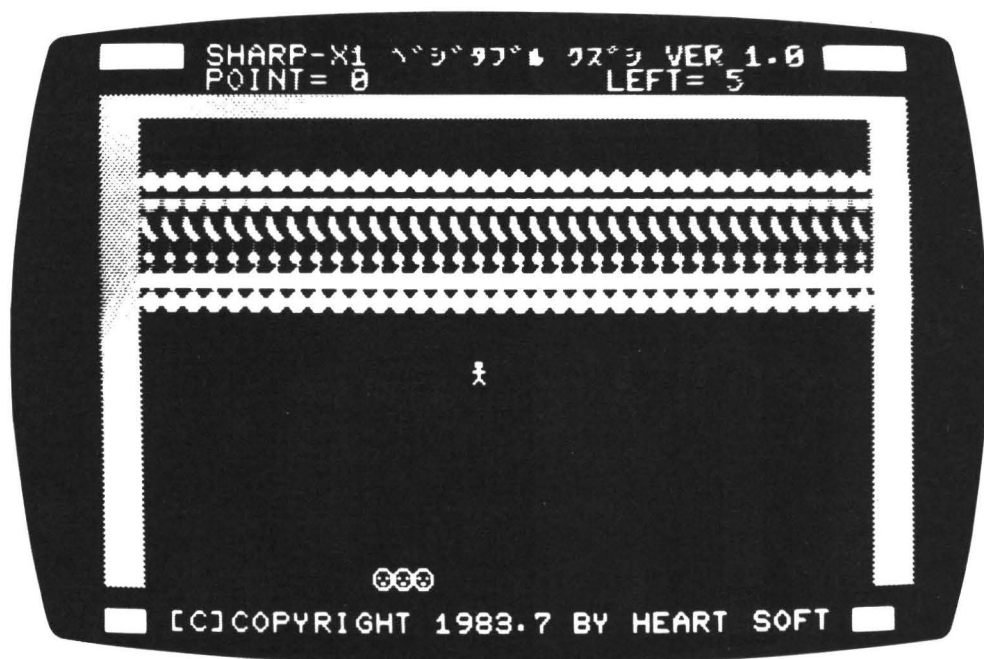
150〜190: 単語のデータを読み込む

350〜410: 文章を単語に分割するルーチン

430〜490: 単語を検索するルーチン

510〜530: バッファのクリアー

550〜620: 単語のデータ

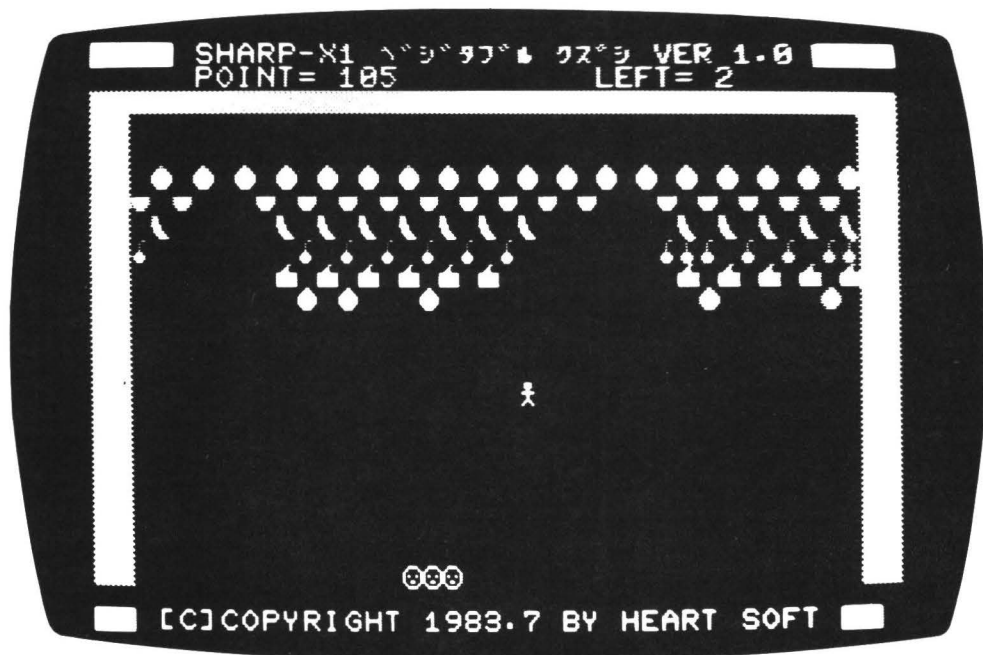
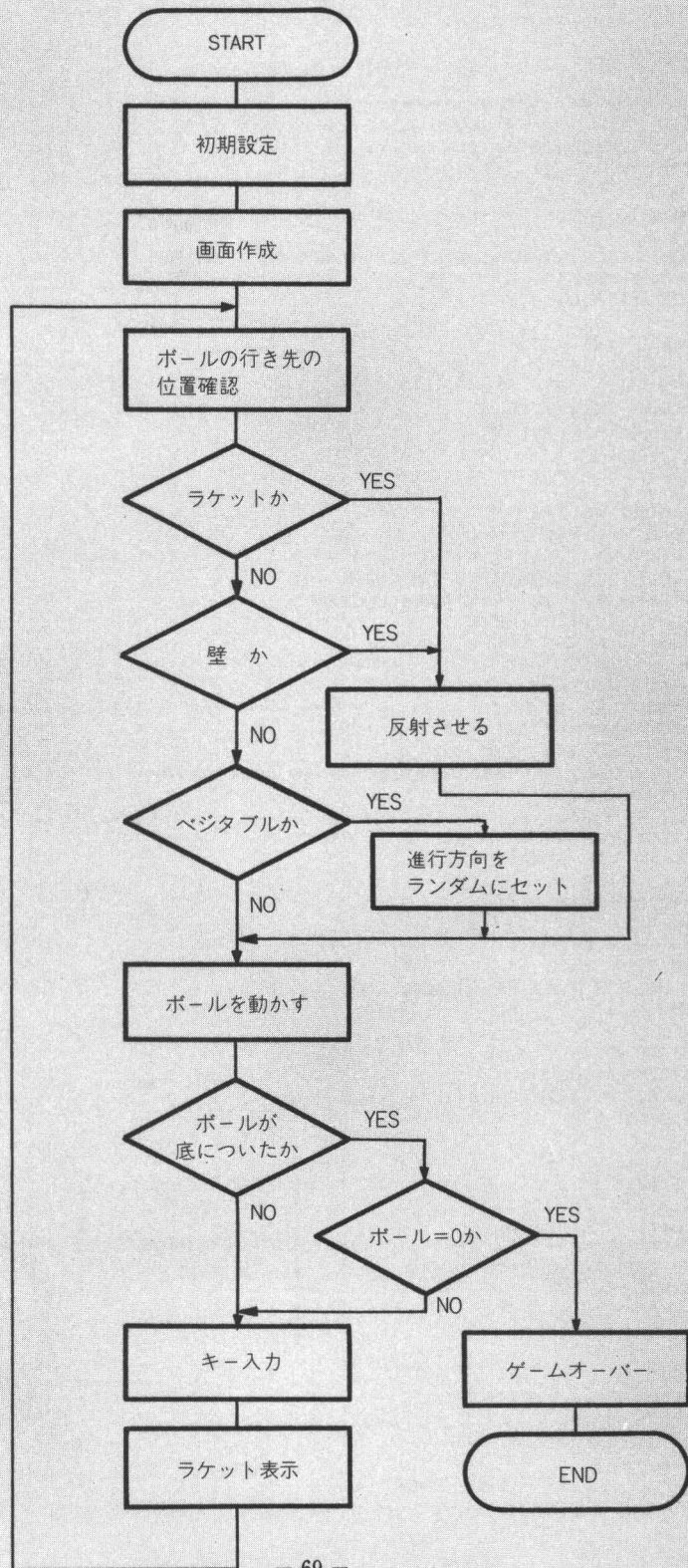
14
ベジタブルくずし

以前インベーダーゲームが出現する前のテーブルゲームの主役は，“ブロックくずし”でした。私も，かなり熱中して財布を軽くしたものです。そこで怨みのゲームをＸ１で作ろうとしたのですが，そのままでは面白くありませんので，そこで，ブロックは色々なキャラクターに作ってみました。一番下から，オレンジ，ショートケーキ，チェリー，バナナ，スイカ，ナシです（一番下はトマト，一番上はグレープフルーツに見える，という意見もあるのですが………）。このデータは本書のキャラクターメーカーで作りましたので，好きなように変更してみて下さい。

ゲームの遊び方

ルールは，ブロックくずしと同じですので簡単です。一応知らない人のために説明しますと，ラケットを左右にうまく動かし，ボールを跳ね返してブロックをこわしていく，というものです。ラケットは，4，6キーで左右に移動させます。ボールは全部で5ヶあります。ラケットはボールより速く移動するようになってますので，あわてないのが高得点へのコツです。

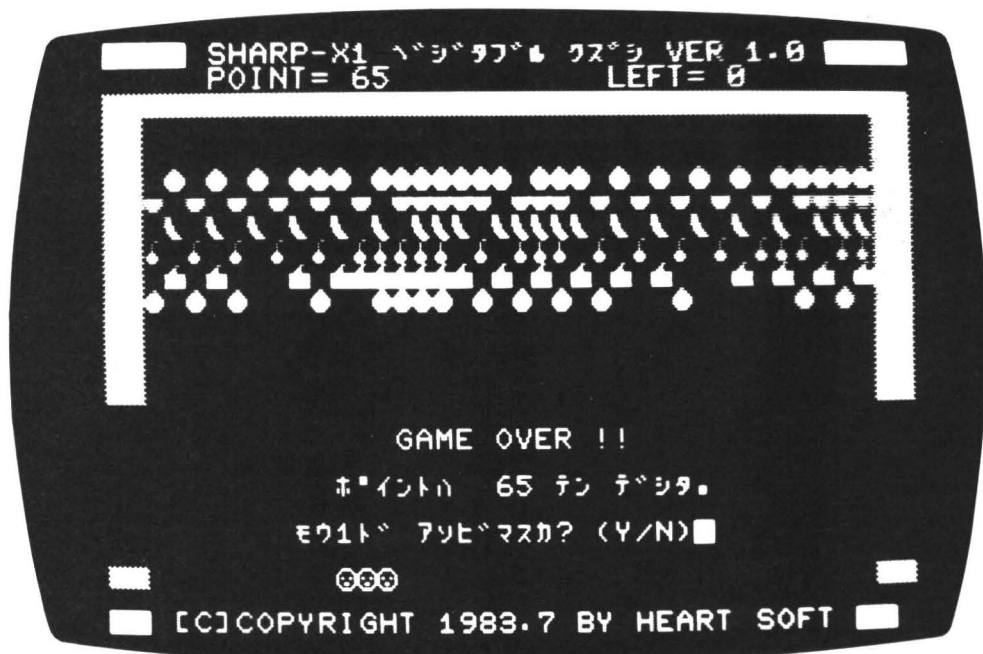
変数表

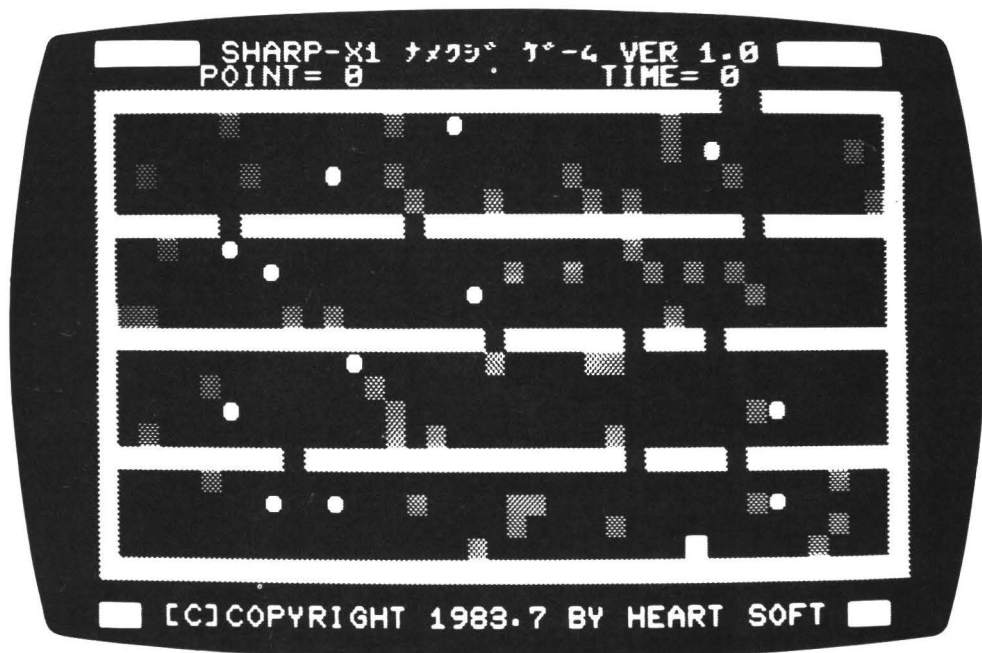
X，Y→ボールのX，Y座標
XX，YY→ボールの進行方向
I＄→キー入力（INKEY＄ で使用）
P＄→画面読み取り
ST→キー入力（STICK 文で使用）
Z→ラケットのX座標
M→残りのボール数
I→ループ用

ベジタブルくずし・フローチャート
START
初期設定
画面作成
ボールの行き先の
位置確認
ラケットか
YES
NO
壁　か
YES
NO
反射させる
ベジタブルか
YES
進行方向を
ランダムにセット
NO
ボールを動かす
ボールが
底についたか
YES
NO
ボール=0か
YES
NO
キー入力
ゲームオーバー
ラケット表示
END

14. ベジタブルくずし

```
1 REM*************************************
2 REM*                                   *
3 REM*      ■■ ﾍﾞｼﾞﾀﾌﾞﾙ ｸｽﾞｼ ■■          *
4 REM*                                   *
5 REM*      [C]Copyright 1983年 7月       *
6 REM*                                   *
7 REM*   FOR SHARP-X1 By Heart Soft      *
8 REM*                                   *
9 REM*************************************
10 CONSOLE 0,25:WIDTH 40:GOSUB 730 ─── データの読み込み
20 CLS:CLICK OFF:COLOR 7:TIME=0          } 初期設定
30 RANDOMIZE:XX=-1:YY=-1:Z=13:M=5        }
40 LOCATE 0,0                                                    }
50 PRINT"■■ SHARP-X1 ﾍﾞｼﾞﾀﾌﾞﾙ ｸｽﾞｼ VER 1.0 ■■";                  }
60 LOCATE 0,24                                                   } 画面出力
70 PRINT"■ [C]COPYRIGHT 1983.7 BY HEART SOFT ■";                 }
80 LOCATE  5,1:PRINT "POINT= 0";                                 }
90 LOCATE 25,1:PRINT "LEFT= 5";                                  }
100 COLOR5                        }
110 LINE(0,22)-(0,2),"▒"          }
120 LINE -(39,2),"▒"              } 枠を作る
130 LINE -(39,22),"▒"             }
140 LINE(1,22)-(1,2),"▒"          }
150 LINE(38,22)-(38,2),"▒"        }
160 COLOR 7
170 GOSUB 430             ──────── ベジタブルを表示
180 XX=-1:YY=-1                          }
190 X=INT(RND(1)*35)+2                   } ボールの出るところを決める
200 Y=INT(RND(1)*10)+5                   }
210 IF CHARACTER$(X,Y)<>" " THEN 190 ─── もしスペースでなければやり直し
220 CGEN1:LOCATE X,Y:PRINT#0 "a";:CGEN 0
230 CGEN 1:LOCATE Z,22:PRINT"■■";:CGEN 0
240 FOR I=1 TO 10:BEEP:NEXT I
250 PAUSE 1
260 LOCATE X,Y:PRINT" ";
270 P$=CHARACTER$(X+XX,Y+YY)
280 IF P$="■" THEN YY=-YY            ──────── ラケットに当った場合
290 IF P$="▒" AND X=2 OR X=37 THEN XX=-XX   } 壁に当った場合
300 IF P$="▒" AND Y=3 THEN YY=-YY           }
310 IF P$<CHR$(230) AND P$>CHR$(224) THEN GOSUB 490 ─── ベジタブルに当った場合
320 X=X+XX:Y=Y+YY
330 IF Y>22 THEN 580                 ──────── ボールを取りそこねたとき
340 CGEN 1:LOCATE X,Y:PRINT#0 "a";:CGEN 0 ─── ボールを表示
350 ST=STICK(0):IF ST=0 THEN 250
360 LOCATE Z,22:PRINT"   ";                 }
370 IF ST=4 THEN Z=Z-1.5                    }
380 IF ST=6 THEN Z=Z+1.5                    } ラケットを動かす
390 IF Z<2  THEN Z=2                        }
400 IF Z>35 THEN Z=35                       }
410 CGEN 1:LOCATE Z,22:PRINT"■■";:CGEN 0    }
420 GOTO 250
430 REM
440 FOR I=0 TO 5                        }
450 LOCATE 2,5+I:CGEN 1                 } ベジタブルを表示する
460 PRINT#0 STRING$(36,CHR$(224+I))     }
470 NEXT I:CGEN 0                       }
480 RETURN
490 REM
500 PO=PO+1:BEEP                        }
510 LOCATE X+XX,Y+YY:PRINT" ";          }
520 LOCATE 11,1:PRINT PO;               }
530 XX=INT(RND(1)*2)                    } ボールがベジタブルに当ったとき
540 IF XX=0 THEN XX=1 ELSE XX=-1        }
550 YY=INT(RND(1)*2)                    }
560 IF YY=0 THEN YY=1 ELSE XX=-1        }
570 RETURN                              }
580 REM
590 GOSUB 820
600 M=M-1:LOCATE 30,1:PRINTM;
610 IF M<>0 THEN 180                ──────── もしボールがあればメインへ
620 LINE(0,15)-(39,21)," ",BF
630 LOCATE 0,16
640 PRINT"               GAME OVER !!
650 PRINT
```

```
660 PRINT"                ﾎﾟｲﾝﾄﾊ ";PO;"ﾃﾝ ﾃﾞｼﾀ｡"      } ゲームオーバー
670 PRINT
680 PRINT"               ﾓｳ1ﾄﾞ ｱｿﾋﾞﾏｽｶ? (Y/N)";
690 I$=INPUT$(1)
700 IF I$="Y" OR I$="y" THEN RUN
710 IF I$="N" OR I$="n" THEN CLS:END
720 GOTO 690
730 DEF CHR$(228)=HEXCHR$("00000000FFFFFF00")+HEXCHR$("001038FFFF00FFFF")+HEXCHR
$("0C0800FFFFFFFFFF")
740 DEF CHR$(226)=HEXCHR$("0000000000000000")+HEXCHR$("006060F070783C1E")+HEXCHR
$("E0E0E0F070783C1E")
750 DEF CHR$(227)=HEXCHR$("0000000000000000")+HEXCHR$("00000000387C7C38")+HEXCHR
$("0C10101000000000")
760 DEF CHR$(224)=HEXCHR$("0000000000000000")+HEXCHR$("002C7EFFFFFF7E3C")+HEXCHR
$("183C7EFFFFFF7E3C")
770 DEF CHR$(229)=HEXCHR$("0000000000000000")+HEXCHR$("002C7EFFFFFF7E3C")+HEXCHR
$("1810000000000000")
780 DEF CHR$(225)=HEXCHR$("0000000000000000")+HEXCHR$("00007E7E7E3C0000")+HEXCHR
$("0000818181423C00")
790 DEF CHR$( 64)=HEXCHR$("3838107C10102844")+HEXCHR$("3838107C10102844")+HEXCHR
$("3838107C10102844")
800 DEF CHR$(135)=HEXCHR$("3C4281A58199423C")+HEXCHR$("3C4281A58199423C")+HEXCHR
$("3C4281A58199423C")
810 RETURN
820 SOUND 7,28:SOUND 1,0:SOUND 3,0
830 FOR I=15 TO 0 STEP -1
840 SOUND 0,255-I*6:SOUND 2,204-I*4:SOUND 8,I:SOUND 9,I
850 SOUND 10,I/1.5:SOUND 6,30-I
860 PAUSE 1
870 NEXT I
880 RETURN
```

ベジタブルのデータ（730〜800行）

得点65でゲーム終了

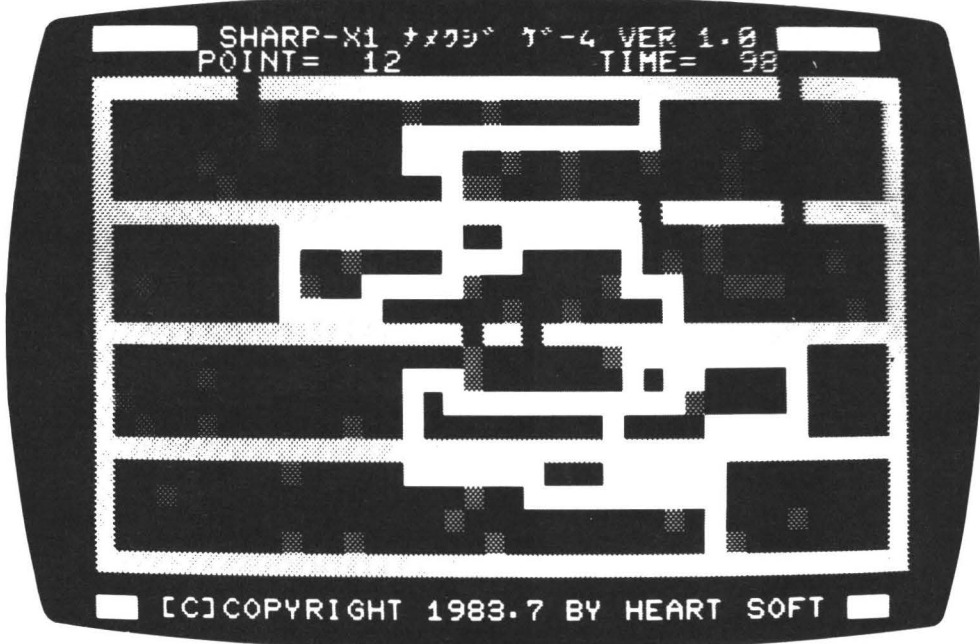
15
ナメクジゲーム

このゲームは最後まで，なんという名前をつけようか悩んだゲームです。先にゲームの構想ができてしまうというのは，私としては，珍しいことなのですが………。そこで，色々とゲーム名を募集したところ，迷コピーライターである上岡君の『まるでナメクジのようだ』との意見が出され，このゲームの名前がナメクジゲームと決まったという次第です，なるほど"■"の通った後が，あのナメクジが，はいずり回った後に良く似ているといえないでもありません。

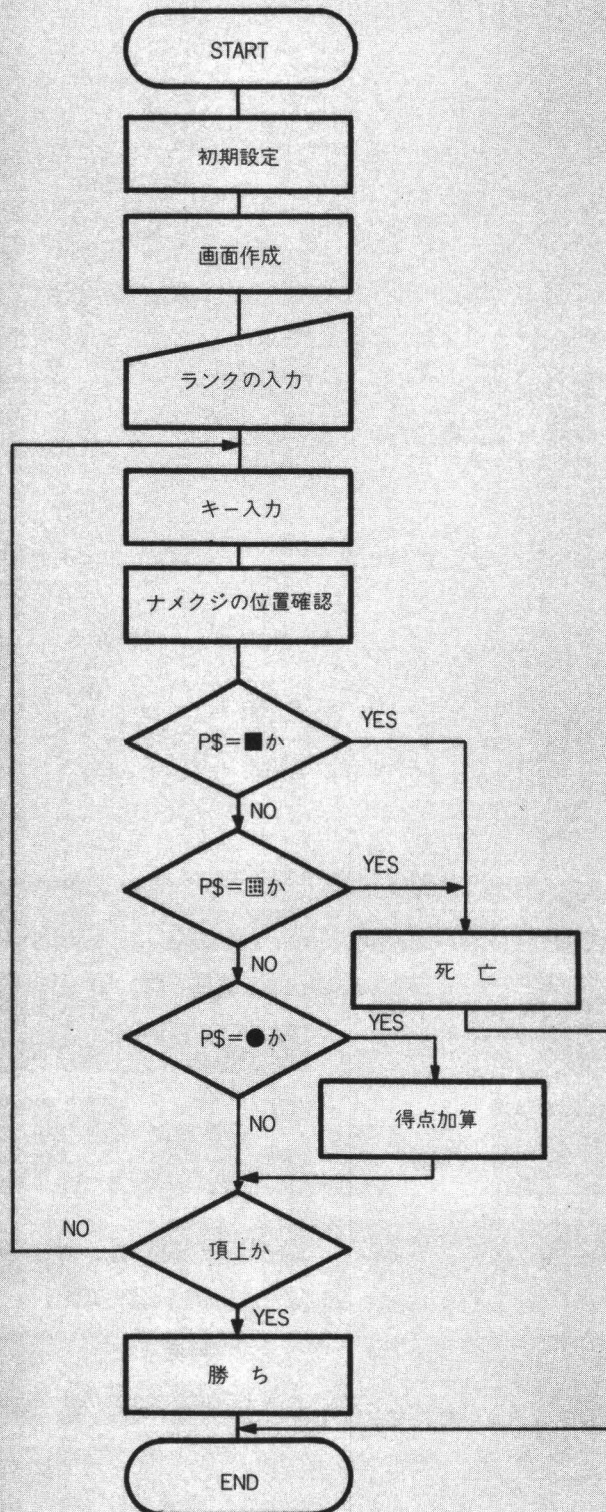
ゲームの遊び方

あなたは〝■〟のナメクジを操って壁に，ぶつからないようにして，いくつのエサを食べることができるか！　というのが，このゲームのストーリーです。

ランクは9コースあり，数字が小さいほどナメクジの進むスピードが速くなります。ナメクジの操作は，テンキー，2→下，4→左，6→右，8→上，です。まず，最下部からスタートしてエサを食べながら，最上部の穴から出ます。壁には，3つの穴がありますので，一回上に昇ってからまた下へ降り，そしてまた上に昇る，ということも可能です。パーフェクトを目ざして頑張って下さい。

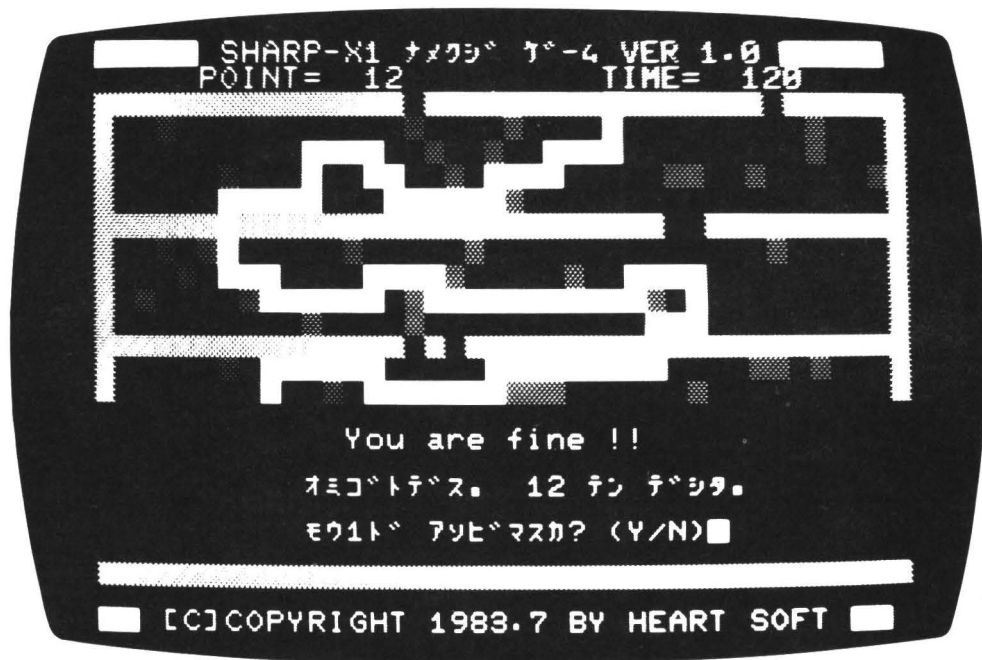
変数表

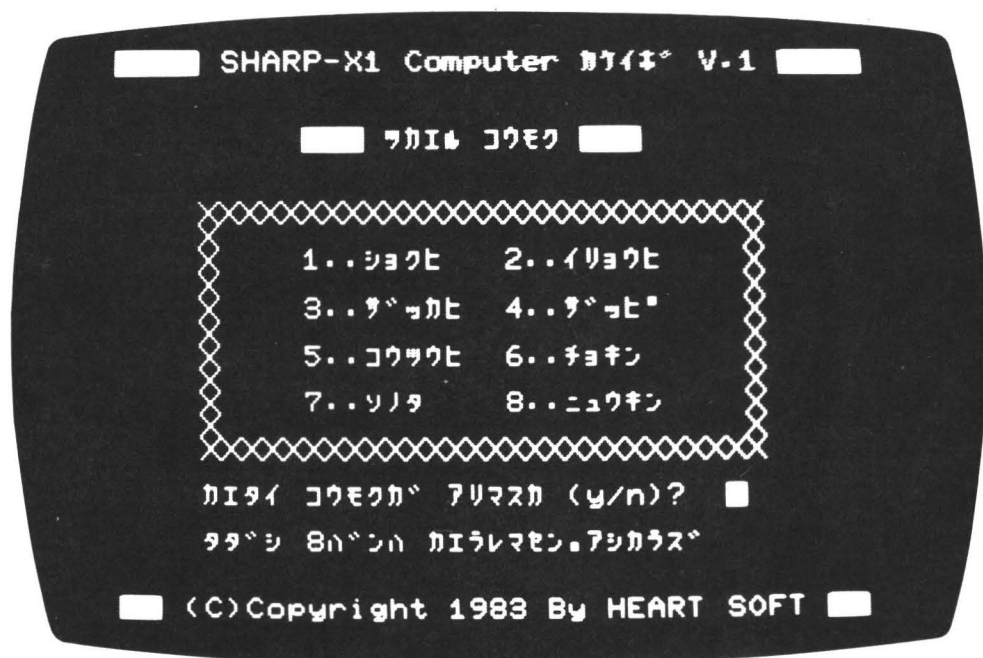
I＄→キー入力用（INPUT＄）
Q→レベル
X，Y→キャラクター表示のX，Y座標
XX，YY→進行方向用のデータ
PO→ポイント
ST→キー入力用（STICK 文）
I，J→ループ用

15. ナメクジゲーム

ナメクジゲーム・フローチャート
START
初期設定
画面作成
ランクの入力
キー入力
ナメクジの位置確認
P$＝■か
YES
NO
P$＝▦か
YES
NO
死　亡
P$＝●か
YES
NO
得点加算
頂上か
NO
YES
勝　ち
END

```
1 REM*************************************
2 REM*                                   *
3 REM*  ■■■■ ﾅﾒｸｼﾞ ｹﾞｰﾑ VER 1.0 ■■■■  *
4 REM*                                   *
5 REM*     [C]Copyright 1983年 7月       *
6 REM*                                   *
7 REM*   FOR SHARP-X1 By Heart Soft      *
8 REM*                                   *
9 REM*************************************
10 CONSOLE 0,25:WIDTH 40
20 CLS:CLICK OFF:COLOR 7,0
30 RANDOMIZE:TIME=0
40 CLS:LOCATE 10,15
50 PRINT"ｱﾅﾀﾉ ﾚﾍﾞﾙ ﾊ ? (1-9)";
60 I$=INPUT$(1)                              レベルの入力　1～9まで
70 IF I$<"1" OR I$>"9" THEN 60
80 Q=VAL(I$)
90 CLS:COLOR 7:LOCATE 0,0
100 PRINT"■■■■ SHARP-X1 ﾅﾒｸｼﾞ ｹﾞｰﾑ VER 1.0 ■■■■";
110 LOCATE 0,24                                              画面表示
120 PRINT"■■ [C]COPYRIGHT 1983.7 BY HEART SOFT ■■";
130 LOCATE 5,1:PRINT"POINT= 0"
140 LOCATE 25,1:PRINT"TIME= 0"
150 COLOR 5
160 LINE(0,2)-(39,22),"▒",B  ―――― 枠を表示する
170 FOR I=1 TO 3
180 LINE(0,I*5+2)-(39,I*5+2),"▒"
190 NEXT I
200 COLOR 4
210 FOR I=1 TO 50
220 X=INT(RND(1)*39)
230 Y=INT(RND(1)*20)+2                        途中に壁を50個表示させる
240 IF CHARACTER$(X,Y)<>" " THEN 220 ――――もしスペースでなければやり直し
250 LOCATE X,Y:PRINT"▒";
260 NEXT I
270 CFLASH 1:COLOR 2
280 FOR I=0 TO 3:FOR J=0 TO 2
290 X=INT(RND(1)*30)+5
300 Y=INT(RND(1)*3)+5*I+3                     ポイントを出力する
310 IF CHARACTER$(X,Y)<>" " THEN 290
320 LOCATE X,Y:PRINT"●";
330 NEXT J:NEXT I
340 CFLASH 0
350 FOR I=0 TO 3:FOR J=0 TO 2
360 X=INT(RND(1)*30)+5
370 LOCATE X,I*5+2:PRINT" ";                  壁をぬける道を作る
380 NEXT J:NEXT I
390 X=INT(RND(1)*30)+5
400 Y=21:COLOR 6
410 IF CHARACTER$(X,Y)<>" " THEN 390          自分の■を表示
420 LOCATE X,Y:PRINT"■";
430 BEEP:PAUSE 30:TEMPO 400
440 ST=STICK(0):PLAY "O4A5B:O4F5G"
450 IF ST=2 THEN XX=0:YY=1
460 IF ST=6 THEN XX=1:YY=0                    KEY入力      STICK文を使用してるので
470 IF ST=8 THEN XX=0:YY=-1                                (1)でジョイスティックになる
480 IF ST=4 THEN XX=-1:YY=0
490 P$=CHARACTER$(X+XX,Y+YY)
500 IF P$="■" OR P$="▒" THEN 650  ―――― 自分か壁に当ったら負け
510 IF P$="●" THEN GOSUB 570      ―――― ポイントを取ったとき
520 IF Y+YY=1 THEN 780
530 X=X+XX:Y=Y+YY:COLOR 6
540 LOCATE X,Y:PRINT"■";
550 PAUSE Q*.5:GOSUB 610          ―――― ランクに応じての時間待ち
560 GOTO 440
570 REM
580 BEEP:PO=PO+1:COLOR 7
590 LOCATE 12,1:PRINT PO;                     ポイントの出力
600 RETURN
610 REM
620 COLOR 7
630 LOCATE 31,1:PRINT TIME                    時間の出力
640 RETURN
```

15. ナメクジゲーム

```
650 REM
660 FOR I=0 TO 10:BEEP:NEXT I
670 LINE(0,15)-(39,21)," ",BF
680 LOCATE 0,16
690 PRINT"                GAME OVER !!
700 PRINT
710 PRINT"          ｱﾅﾀﾉ ﾎﾟｲﾝﾄﾊ ";PO;"ﾃﾝ ﾃﾞｼﾀ｡"
720 PRINT
730 PRINT"            ﾓｳ1ﾄﾞ ｱｿﾋﾞﾏｽｶ? (Y/N)";
740 I$=INPUT$(1)
750 IF I$="Y" OR I$="y" THEN RUN
760 IF I$="N" OR I$="n" THEN CLS:END
770 GOTO 740
780 REM
790 TEMPO 300
800 FOR I=0 TO 30:PLAY "O5G0B:O5B0O6D":NEXT
810 LINE(0,15)-(39,21)," ",BF
820 LOCATE 0,16
830 PRINT"               You are fine !!
840 PRINT
850 PRINT"            ｵﾐｺﾞﾄﾃﾞｽ｡ ";PO;"ﾃﾝ ﾃﾞｼﾀ｡"
860 PRINT
870 PRINT"            ﾓｳ1ﾄﾞ ｱｿﾋﾞﾏｽｶ? (Y/N)";
880 I$=INPUT$(1)
890 IF I$="Y" OR I$="y" THEN RUN
900 IF I$="N" OR I$="n" THEN CLS:END
910 GOTO 880
```

（650〜770）負けのメッセージ
（780〜910）勝ちのメッセージ

お見事！得点12でゲーム終了．
もっと短時間で終了するよう頑張ろう．

16
コンピューター家計簿

マイコンをゲームマシンにしては勿体無い限りです。そこで一番身近で実用になるものを，ということで作ったのが，この“コンピューター家計簿”です。

本書の中のプログラムにしては，わりと長く，リストをプリンターで出力すると，プリンター用紙では7枚にわたってしまいます。まあ，気長に，ゆっくりと打ち込んで下さい。以下に使用法を書きますが，ただ読んだだけでは分かりにくいと思います。まずは使用してみて下さい。使ってみれば，さほど難しいことではないと思います。

プログラムの使用法

プログラムをRUNさせると，まずこの家計簿で使える項目名が表示されます。ここで項目を変更するか尋ねてきます。項目は全部で8項目ありますが，最後の“入金”だけは変更できないようになっています。これは，この後の説明のコマンド7で，グラフを出力する場合に使用しているためです。項目を変更するときは〝y〟を，しないときは〝n〟を押して下さい。yを押したときは，変更したい項目ナンバー，項目名を，画面の手順にそって入力して下さい。この処理を終えると，次にコマンドのメニューが表示されます。コマンドは全部で8コマンド有り，1～3までが出力関係，4～5までが入力関係，6～8まではその他です。このコマンド・メニューが，メインルーチンになっていますので，各処理を終えると，ここへ戻って来るようになっています。

☆コマンド1

このコマンドは，内部の数値をカセットテープに出力するもので，データの保存用として使用します。画面にメッセージが出ますので，新しいテープをセットして RETURN キーを押して下さい。メッセージと共にデータが出力されます。X1は，カセットのデータ出力スピードが速いため，特にテープエラーを起こすことがありますので，データは，なるべく2回以上，保存するようにして下さい。だいたい，30秒～1分ほどで終ります。

☆コマンド2

このコマンドは，内部の数値をプリンターに出力するものです。80桁が出力できるプリンターならば，どのプリンターでも構わないのですが，シャープX1純正以外のプリンターは，キャラクターコードの違いに気をつけて下さい。

☆コマンド3

このコマンドは，内部の数値をテレビ画面に出力するものです。画面上1度に，1ヶ月分31日は出力できませんので，10日間ずつ区切って出力します。最下部には累積の合計が出力されます。

☆コマンド4

このコマンドは，内部へ数値をカセットテープから入力するもので，丁度コマンド1の逆に当ります。使用法も，コマンド1と同じで，メッセージの後に，今度はコマンド1で作ったテープをセットし RETURN キーを押します。

☆コマンド5

このコマンドは，内部へ数値を直接キーボードから入力するものです。まず入力する月，日を入力して RETURN キーを押すと，画面にグラフと数値，項目名が出力されます。データを変更するか尋ねてきますが，最初はすべてが0なので，初めは“y”と入力して下さい。するとカーソルが左上に移りますので，後は好きなようにカーソルコントロールキーを使って数値を打ち込んで下さい。数字以外のものを入力すると，0と判断します。1項目につき，4データ8桁まで入力可能です。入力が終ったら RETURN キーを押して下さい。間違いがないかもう一度見直して，よければ，もう一度 RETURN キーを押して下さい。自動的にデータを読み込んでくれます。更にまた変更するか尋ねてき，“n”を押すと，最後に次の日の分も入力するか尋ねてきます。yを押せば，また次の日も同じように入力して下さい。nを押せば，これでメニューに戻ります。

☆コマンド6

このコマンドは，プログラムを開始したときに出て来る，項目名の変更と同じものです。操作も前記の通りで，最初に変更しそこねたときに使用して下さい。

☆コマンド7

このコマンドは，内部の数値をグラフにして出力するものです。グラフは全部で3本出力されます。1番上は1ヶ月の出金状態を表示したもので，これは百分率で表わされ，4ドットが10%に当ります。2番，3番目のグラフは，入金と出金の関係を表示したもので，どちらか多い方を 100%としています。

☆コマンド8

このコマンドは，このプログラムを止めるときに使用します。もし，間違ってこのコマンドを使ったときや，BREAK キーを押してしまったときは，直接 GOTO 90 [CR] として下さい。内部データは保存されます。

変数表

A＃(n，m)→31日分8項目の数値

K＄(n)→項目のデータ

X＃(n，m)→1日分8項目の数値

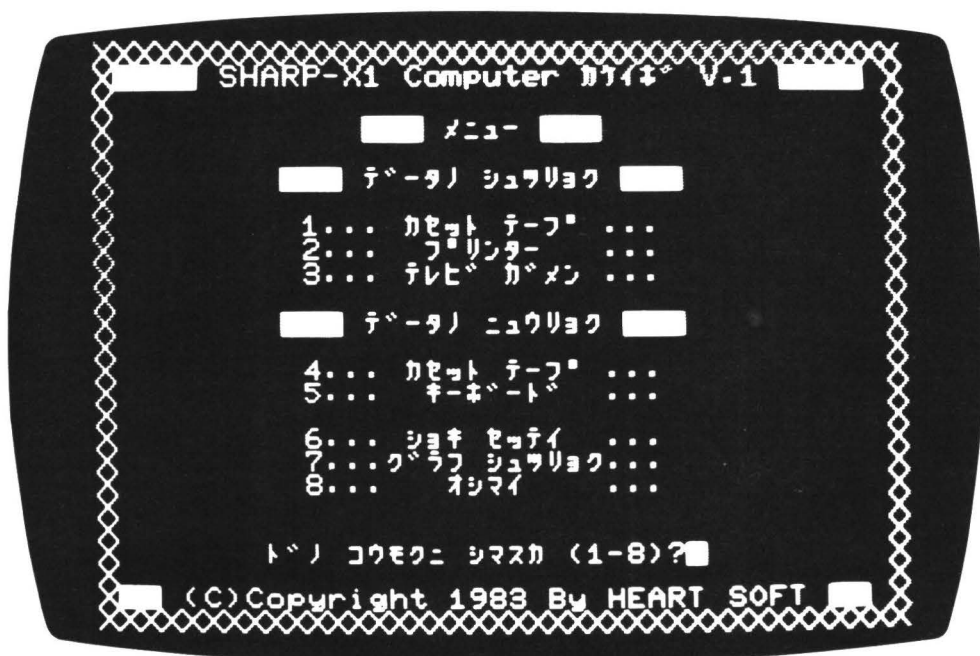
I＄(n)→画面入力用 LINEINPUT 文で使用

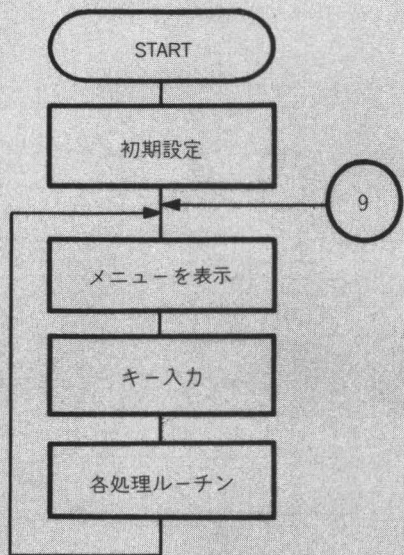
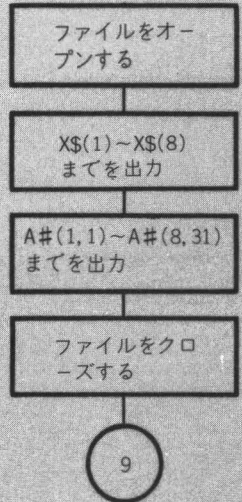
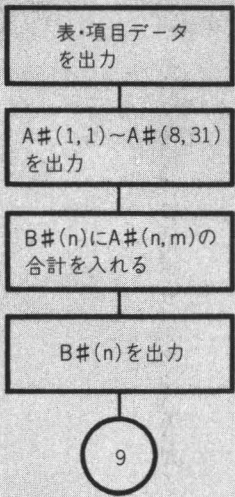
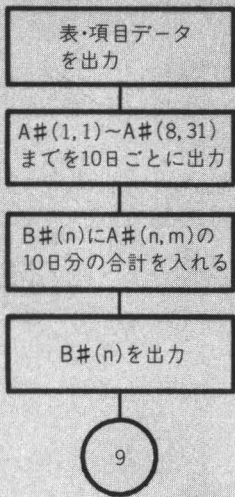
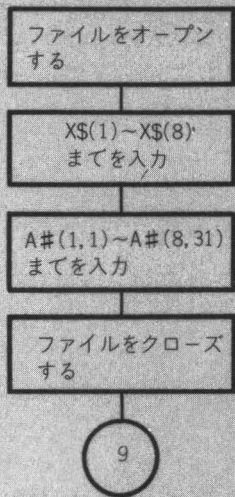
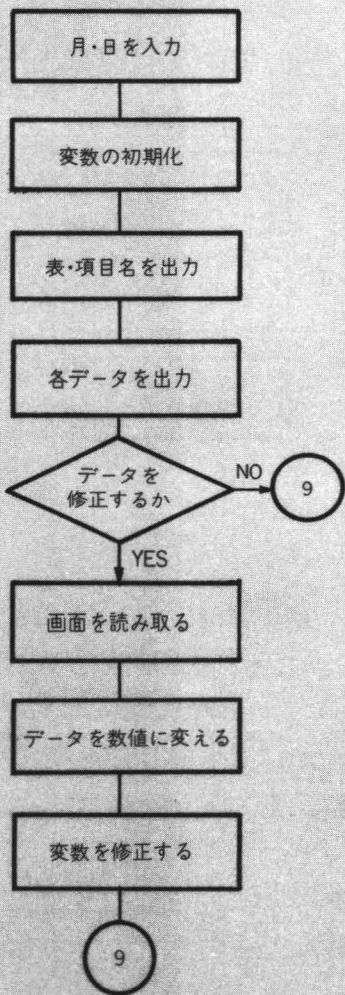
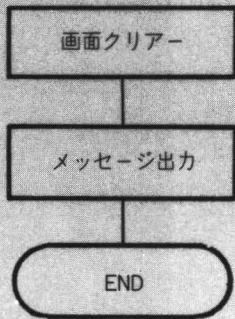
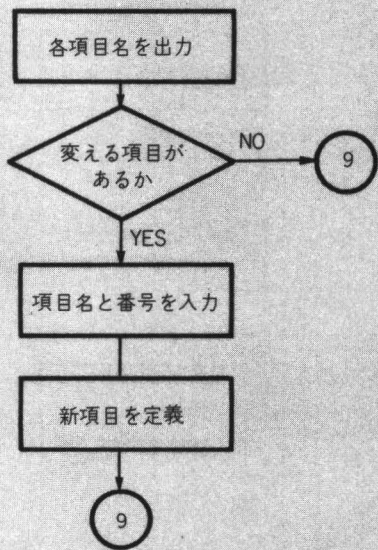
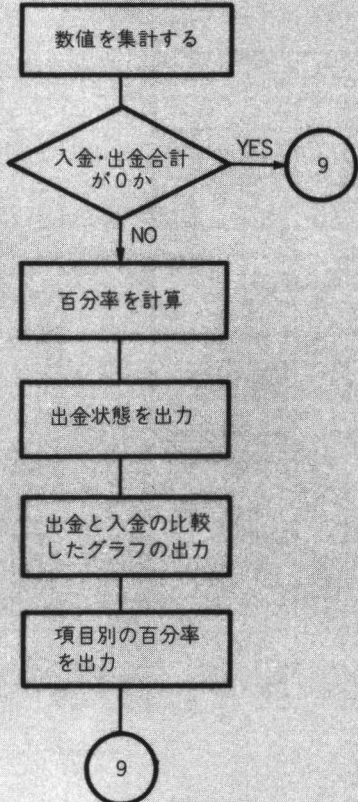
B＃(n)→計算用A＃（n，1～31）まで

R(n)→各データの百分率

M→月

P→日

I，J→ループ用

コンピュータ家計簿のメニュー画面

家計簿・フローチャート(1)
START
初期設定
9
メニューを表示
キー入力
各処理ルーチン
① カセットへの出力
ファイルをオープンする
X$(1)~X$(8)までを出力
A#(1,1)~A#(8,31)までを出力
ファイルをクローズする
9
② プリンターへの出力
表・項目データを出力
A#(1,1)~A#(8,31)を出力
B#(n)にA#(n,m)の合計を入れる
B#(n)を出力
9
③ 画面への出力
表・項目データを出力
A#(1,1)~A#(8,31)までを10日ごとに出力
B#(n)にA#(n,m)の10日分の合計を入れる
B#(n)を出力
9
④ カセットからの入力
ファイルをオープンする
X$(1)~X$(8)までを入力
A#(1,1)~A#(8,31)までを入力
ファイルをクローズする
9

家計簿・フローチャート(2)
⑤ キーボードからの入力
月・日を入力
変数の初期化
表・項目名を出力
各データを出力
データを修正するか
NO
9
YES
画面を読み取る
データを数値に変える
変数を修正する
9
⑥ 初期設定
各項目名を出力
変える項目があるか
NO
9
YES
項目名と番号を入力
新項目を定義
9
⑦ グラフ出力
数値を集計する
入金・出金合計が0か
YES
9
NO
百分率を計算
出金状態を出力
出金と入金の比較したグラフの出力
項目別の百分率を出力
9
⑧ 終り
画面クリアー
メッセージ出力
END

```
1 REM**************************************
2 REM*                                    *
3 REM*  ■■ Computer ｶｹｲﾎﾞ Ver 1.0 ■■     *
4 REM*                                    *
5 REM*      [C]Copyright 1983年 7月       *
6 REM*                                    *
7 REM*   For SHARP-X1 By HEART SOFT       *
8 REM*                                    *
9 REM**************************************
10 CONSOLE 0,25:WIDTH 40
20 DIM A#(8,31),K$(8),X#(8,4),I$(8),B#(8),R(8)
30 K$(1)="ｼｮｸﾋ":K$(2)="ｲﾘｮｳﾋ":K$(3)="ｻﾞｯｶﾋ"
40 K$(4)="ｻﾞｯﾋﾟ":K$(5)="ｺｳﾂｳﾋ":K$(6)="ﾁｮｷﾝ"
50 K$(7)="ｿﾉﾀ":K$(8)="ﾆｭｳｷﾝ":M=0:P=1
60 GOTO 1690
70 REM ■■ ﾒｲﾝ ﾙｰﾁﾝ ■■
80 CONSOLE 0,25
90 WIDTH 40:PRINT CHR$(12);
100 LOCATE 0,1
110 PRINT " ■■ SHARP-X1 Computer ｶｹｲﾎﾞ V.1 ■■";
120 LOCATE 0,23
130 PRINT " ■■ (C)Copyright 1983 By HEART SOFT ■■";
140 LOCATE 13,3
150 PRINT"■■ ﾒﾆｭｰ ■■"
160 LOCATE 0,5
170 PRINT TAB(9);"■■ ﾃﾞｰﾀﾉ ｼｭﾂﾘｮｸ ■■ "
180 PRINT
190 PRINT TAB(10);"1... ｶｾｯﾄ ﾃｰﾌﾟ ..."
200 PRINT TAB(10);"2...  ﾌﾟﾘﾝﾀｰ   ..."
210 PRINT TAB(10);"3... ﾃﾚﾋﾞ ｶﾞﾒﾝ ..."
220 PRINT
230 PRINT TAB(9);"■■ ﾃﾞｰﾀﾉ ﾆｭｳﾘｮｸ ■■ "
240 PRINT
250 PRINT TAB(10);"4... ｶｾｯﾄ ﾃｰﾌﾟ ..."
260 PRINT TAB(10);"5...   ｷｰﾎﾞｰﾄﾞ   ..."
270 PRINT
280 PRINT TAB(10);"6... ｼｮｷ ｾｯﾃｲ  ..."
290 PRINT TAB(10);"7...ｸﾞﾗﾌ ｼｭﾂﾘｮｸ..."
300 PRINT TAB(10);"8...    ｵｼﾏｲ    ..."
310 LINE(0,0)-(39,24),"X",B
320 LOCATE 8,21
330 PRINT"ﾄﾞﾉ ｺｳﾓｸﾆ ｼﾏｽｶ (1-8)?";
340 I$=INPUT$(1)
350 IF I$<"1" OR I$>"8" THEN 340
360 ON VAL(I$) GOTO 370,600,900,1460,2110,1690,3110,3570
370 REM ■■ ｶｾｯﾄﾍﾉ ｼｭﾂﾘｮｸ ■■
380 PRINT CHR$(12);
390 LOCATE 0,1
400 PRINT " ■■ SHARP-X1 Computer ｶｹｲﾎﾞ V.1 ■■";
410 LOCATE 0,23
420 PRINT " ■■ (C)Copyright 1983 By HEART SOFT ■■";
430 LINE(0,0)-(39,24),"X",B
440 LOCATE 8,10
450 PRINT "ｱﾀﾗｼｲ ｶｾｯﾄﾃｰﾌﾟｦ ｲﾚﾃｸﾀﾞｻｲ｡"
460 LOCATE 5,12
470 PRINT "ﾖｳｲｶﾞ ﾃﾞｷﾏｼﾀﾗ RETURNｦ ｵｼﾃｸﾀﾞｻｲ｡";
480 I$=INPUT$(1)
490 IF I$<>CHR$(13) THEN 480
500 OPEN"O",#1,"ｶｹｲﾎﾞﾉ ﾃﾞｰﾀ"
510 FOR I=1 TO 8:PRINT #1,X$(I)
520 NEXT I
530 FOR I=1 TO 31
540 LOCATE 5,15
550 PRINT "ﾀﾀﾞｲﾏ";I;"日ﾌﾞﾝﾉ ﾃﾞｰﾀｦ ｶｷｺﾐﾁｭｳ ﾃﾞｽ｡";
560 PRINT#1,A#(1,I),A#(2,I),A#(3,I),A#(4,I),A#(5,I),A#(6,I),A#(7,I),A#(8,I)
570 NEXT I
580 CLOSE #1
590 GOTO 70
600 REM■■ ﾌﾟﾘﾝﾀｰﾍﾉ ｼｭﾂﾘｮｸ ■■
610 FOR I=0 TO 8:B#(I)=0:NEXT I
620 LPRINTTAB(20);
630 LPRINT " ■■ SHARP-X1 Computer ｶｹｲﾎﾞ V.1 ■■";
640 LPRINT:LPRINT
650 LPRINT"  ┌────┬────┬────┬────┬────┬────┬────┬────
────┐"
```

- 30〜50: 項目のデータ
- 140〜290: メニューを表示
- 320〜360: キー入力1〜8まで
- 440〜470: メッセージ出力
- 500: ファイルをオープンする
- 510〜520: 項目のデータを出力
- 560: 31日分各データの出力
- 580: ファイルをクローズする

16. コンピューター家計簿

```
660 LPRINT"  | 日|"RIGHT$("          "+K$(1),8)"|"RIGHT$("          "+K$(2),8)"|"RIGH
T$("          "+K$(3),8)"|"RIGHT$("          "+K$(4),8)"|"RIGHT$("          "+K$(5),8)
"|"RIGHT$("          "+K$(6),8)"|"RIGHT$("          "+K$(7),8)"|"RIGHT$("          "+K
$(8),8)"|"
670 LPRINT"  ├──┼────────┼────────┼────────┼────────┼────────┼────────┼────────┼
────────┤"
680 FOR I=1 TO 31
690 LPRINT"  |";:LPRINT USING"##";I; ── 日を表示
700 LPRINT"|";
710 FOR J=1 TO 8
720 LPRINT USING"########";A#(J,I); ── 数値を表示
730 LPRINT"|";
740 NEXT J:LPRINT:NEXT I
750 LPRINT"  ├──┼────────┼────────┼────────┼────────┼────────┼────────┼────────┼
────────┤"
760 FOR I=1 TO 8
770 FOR J=1 TO 31
780 B#(I)=B#(I)+A#(I,J)
790 NEXT J:NEXT I
800 LPRINT"  | =|";
810 FOR I=1 TO 8
820 LPRINT USING"########";B#(I); ── 合計の数値を表示
830 LPRINT "|";
840 NEXT I:LPRINT
850 LPRINT"  └──┴────────┴────────┴────────┴────────┴────────┴────────┴────────┴
────────┘"
860 LPRINT
870 LPRINTTAB(20);
880 PRINT " ■■ (C)Copyright 1983 By HEART SOFT ■■";
890 GOTO 70
900 REM ■■■ ｶﾞﾒﾝﾍﾉ ｼｭﾂﾘｮｸ ■■■
910 WIDTH 80
920 FOR I=0 TO 8:B#(I)=0:NEXT I:GOTO 1170
930 PRINT CHR$(12);
940 LOCATE 20,1
950 PRINT " ■■■■ SHARP-X1 Computer ｶｹｲﾎﾞ V.1 ■■■■";
960 LOCATE 20,23
970 PRINT " ■■ (C)Copyright 1983 By HEART SOFT ■■";
980 LOCATE 0,4
990 PRINT"  ┌──┬────────┬────────┬────────┬────────┬────────┬────────┬────────┬
────────┐"
1000 PRINT"  | 日|"RIGHT$("          "+K$(1),8)"|"RIGHT$("          "+K$(2),8)"|"RIGH
T$("          "+K$(3),8)"|"RIGHT$("          "+K$(4),8)"|"RIGHT$("          "+K$(5),8)
"|"RIGHT$("          "+K$(6),8)"|"RIGHT$("          "+K$(7),8)"|"RIGHT$("          "+K
$(8),8)"|"
1010 PRINT"  ├──┼────────┼────────┼────────┼────────┼────────┼────────┼────────┼
────────┤"
1020 PRINT"  |  |        |        |        |        |        |        |        |
        |"
1030 PRINT"  |  |        |        |        |        |        |        |        |
        |"
1040 PRINT"  |  |        |        |        |        |        |        |        |
        |"
1050 PRINT"  |  |        |        |        |        |        |        |        |
        |"
1060 PRINT"  |  |        |        |        |        |        |        |        |
        |"
1070 PRINT"  |  |        |        |        |        |        |        |        |
        |"
1080 PRINT"  |  |        |        |        |        |        |        |        |
        |"
1090 PRINT"  |  |        |        |        |        |        |        |        |
        |"
1100 PRINT"  |  |        |        |        |        |        |        |        |
        |"
1110 PRINT"  |  |        |        |        |        |        |        |        |
        |"
1120 PRINT"  |  |        |        |        |        |        |        |        |
        |"
1130 PRINT"  ├──┼────────┼────────┼────────┼────────┼────────┼────────┼────────┼
────────┤"
1140 PRINT"  | =|        |        |        |        |        |        |        |
        |"
```

項目のデータをそろえて表示 (lines 660)

```
1150 PRINT"    └──┴──────┴──────┴──────┴──────┴──────┴──────┴──────┴──────┘
────────┘"
1160 RETURN
1170 REM ■■■■ ﾃﾞｰﾀﾉ ﾌﾟﾘﾝﾄ ■■■■
1180 FOR K=0 TO 2
1190 GOSUB 930
1200 FOR I=1+K*10 TO K*10+11
1210 LOCATE 3,I+6-K*10:PRINT USING"##";I ―――― 日数を表示
1220 NEXT I
1230 FOR I=1+K*10 TO K*10+11                                   ┐
1240 LOCATE  6,I+6-K*10:PRINT USING"########";A#(1,I)          │
1250 LOCATE 15,I+6-K*10:PRINT USING"########";A#(2,I)          │
1260 LOCATE 24,I+6-K*10:PRINT USING"########";A#(3,I)          │
1270 LOCATE 33,I+6-K*10:PRINT USING"########";A#(4,I)          ├ 各項目の数値を出力
1280 LOCATE 42,I+6-K*10:PRINT USING"########";A#(5,I)          │
1290 LOCATE 51,I+6-K*10:PRINT USING"########";A#(6,I)          │
1300 LOCATE 60,I+6-K*10:PRINT USING"########";A#(7,I)          │
1310 LOCATE 69,I+6-K*10:PRINT USING"########";A#(8,I)          ┘
1320 NEXT I
1330 FOR I=1 TO 8
1340 FOR J=1+K*10 TO K*10+11
1350 B#(I)=B#(I)+A#(I,J)        ―――――――――― 数値の合計を計算する
1360 NEXT J:NEXT I
1370 FOR I=0 TO 7
1380 LOCATE 6+I*9,19:PRINT USING"########";B#(I+1)―― 合計を表示
1390 NEXT I
1400 LOCATE 25,21                                    ┐
1410 PRINT "ﾖﾐｵﾜｯﾀﾗ SPACEｷｰｦ ｵｼﾃ ｸﾀﾞｻｲ｡";            │
1420 I$=INPUT$(1)                                    ├ キー入力
1430 IF I$<>" " THEN 1420                            │
1440 NEXT K                                          ┘
1450 GOTO 70
1460 REM ■■■■ ｶｾｯﾄ ｶﾗﾉ ﾆｭｳﾘｮｸ ■■■■
1470 PRINT CHR$(12);
1480 LOCATE 0,1
1490 PRINT " ■■■■ SHARP-X1 Computer ｶｹｲﾎﾞ V.1 ■■■■";
1500 LOCATE 0,23
1510 PRINT " ■■■ (C)Copyright 1983 By HEART SOFT ■■■";
1520 LINE(0,0)-(39,24),"X",B
1530 LOCATE 4,10
1540 PRINT "ﾃﾞｰﾀﾉ ﾊｲｯﾃｲﾙ ｶｾｯﾄﾃｰﾌﾟｦ ｲﾚﾃｸﾀﾞｻｲ｡"
1550 LOCATE 5,12
1560 PRINT "ﾖｳｲｶﾞ ﾃﾞｷﾏｼﾀﾗ RETURNｦ ｵｼﾃｸﾀﾞｻｲ｡";
1570 I$=INPUT$(1)
1580 IF I$<>CHR$(13) THEN 1570
1590 OPEN"I",#1,"ｶｹｲﾎﾞﾉ ﾃﾞｰﾀ"
1600 FOR I=1 TO 8:INPUT #1,X$(I)
1610 NEXT I
1620 FOR I=1 TO 31
1630 LOCATE 5,15
1640 PRINT "ﾀﾀﾞｲﾏ";I;"日ﾌﾞﾝﾉ ﾃﾞｰﾀｦ ﾖﾐｺﾐﾁｭｳ ﾃﾞｽ｡";
1650 INPUT#1,A#(1,I),A#(2,I),A#(3,I),A#(4,I),A#(5,I),A#(6,I),A#(7,I),A#(8,I)
1660 NEXT I
1670 CLOSE #1
1680 GOTO 70
1690 REM ■■■■ ｼｮｷ ｾｯﾃｲ ■■■■
1700 PRINT CHR$(12);
1710 PRINT " ■■■■ SHARP-X1 Computer ｶｹｲﾎﾞ V.1 ■■■■";
1720 LOCATE 0,23
1730 PRINT " ■■■ (C)Copyright 1983 By HEART SOFT ■■■";
1740 LOCATE 10,3
1750 PRINT "■■■■ ﾂｶｴﾙ ｺｳﾓｸ ■■■■"
1760 LOCATE 10,8                  ┐
1770 PRINT "1..";K$(1)            │
1780 LOCATE 20,8                  │
1790 PRINT "2..";K$(2)            │
1800 LOCATE 10,10                 │
1810 PRINT "3..";K$(3)            │
1820 LOCATE 20,10                 │
1830 PRINT "4..";K$(4)            ├ 各項目名を出力
1840 LOCATE 10,12                 │
1850 PRINT "5..";K$(5)            │
1860 LOCATE 20,12                 │
1870 PRINT "6..";K$(6)            ┘
```

```
1880 LOCATE 10,14
1890 PRINT "7.. ";K$(7)
1900 LOCATE 20,14
1910 PRINT "8.. ";K$(8)
1920 LOCATE 5,18
1930 PRINT "ｶｴﾀｲ ｺｳﾓｸｶﾞ ｱﾘﾏｽｶ (y/n)?"
1940 LOCATE 5,20
1950 PRINT "ﾀﾀﾞｼ 8ﾊﾞﾝﾊ ｶｴﾗﾚﾏｾﾝ｡ｱｼｶﾗｽﾞ"
1960 LINE(5,6)-(32,16),"X",B
1970 LOCATE 31,18
1980 I$=INPUT$(1)
1990 IF I$<>"y" THEN 70
2000 LOCATE 0,18
2010 PRINT SPACE$(120)
2020 LOCATE 5,18                                     } メッセージ出力&キー入力
2030 PRINT "ﾄﾞﾉ ｺｳﾓｸｦ ﾍﾝｺｳ ｼﾏｽｶ (1-7)";
2040 I$=INPUT$(1)
2050 IF I$<"1" OR I$>"7" THEN 2040
2060 I=VAL(I$):PRINT I
2070 LOCATE 5,20
2080 PRINT "ｺｳﾓｸｦ ﾆｭｳﾘｮｸ ｼﾃｸﾀﾞｻｲ";
2090 INPUT I$:K$(I)=I$
2100 GOTO 1690
2110 REM ████ ｷｰﾎﾞｰﾄﾞ ｶﾗﾉ ﾆｭｳﾘｮｸ ████
2120 PRINT CHR$(12);
2130 LOCATE 0,18
2140 PRINT"  ﾃﾞｰﾀｦ ﾆｭｳﾘｮｸｽﾙ 月ﾄ 日ｦ ﾆｭｳﾘｮｸｼﾃ ｸﾀﾞｻｲ｡"
2150 LOCATE 5,20
2160 PRINT"月月,日日ﾉﾖｳﾆ ｼﾃｸﾀﾞｻｲ｡";
2170 INPUT M,P
2180 FOR I=0 TO 8:FOR J=0 TO 4:X#(I,J)=0:NEXT J:NEXT I ——— 変数クリアー
2190 PRINT CHR$(12);
2200 LOCATE 0,0
2210 PRINT " ████ SHARP-X1 Computer ｶｹｲﾎﾞ V.1 ████";
2220 LOCATE 0,24
2230 PRINT " ██ (C)Copyright 1983 By HEART SOFT ██";
2240 LOCATE 25,1
2250 PRINT M;"月";P;"日"
2260 LOCATE 0,2
2270 PRINT"┌┬────────┬────────┬────────┬────────┐"
2280 PRINT"| |";RIGHT$("        "+K$(1),8);"|";RIGHT$("        "+K$(2),8);"|";RI
GHT$("        "+K$(3),8);"|";RIGHT$("        "+K$(4),8);"|"
2290 PRINT"├┼────────┼────────┼────────┼────────┤"
2300 PRINT"|1|        |        |        |        |"
2310 PRINT"|2|        |        |        |        |"
2320 PRINT"|3|        |        |        |        |"
2330 PRINT"|4|        |        |        |        |"
2340 PRINT"├┼────────┼────────┼────────┼────────┤"
2350 PRINT"|=|        |        |        |        |"
2360 PRINT"├┼────────┼────────┼────────┼────────┤"
2370 PRINT"| |";RIGHT$("        "+K$(5),8);"|";RIGHT$("        "+K$(6),8);"|";RI
GHT$("        "+K$(7),8);"|";RIGHT$("        "+K$(8),8);"|"
2380 PRINT"├┼────────┼────────┼────────┼────────┤"
2390 PRINT"|1|        |        |        |        |"
2400 PRINT"|2|        |        |        |        |"
2410 PRINT"|3|        |        |        |        |"
2420 PRINT"|4|        |        |        |        |"
2430 PRINT"├┼────────┼────────┼────────┼────────┤"
2440 PRINT"|=|        |        |        |        |"
2450 PRINT"└┴────────┴────────┴────────┴────────┘"
2460 FOR I=1 TO 4
2470 LOCATE 3 ,I+4:PRINT USING"########";X#(1,I)
2480 LOCATE 12,I+4:PRINT USING"########";X#(2,I)
2490 LOCATE 21,I+4:PRINT USING"########";X#(3,I)
2500 LOCATE 30,I+4:PRINT USING"########";X#(4,I)
2510 NEXT I
2520 LOCATE 3 ,10:PRINT USING"########";A#(1,P)
2530 LOCATE 12,10:PRINT USING"########";A#(2,P)
2540 LOCATE 21,10:PRINT USING"########";A#(3,P)
2550 LOCATE 30,10:PRINT USING"########";A#(4,P)
2560 FOR I=1 TO 4                                    } 各データを出力する
2570 LOCATE 3 ,I+13:PRINT USING"########";X#(5,I)
2580 LOCATE 12,I+13:PRINT USING"########";X#(6,I)
2590 LOCATE 21,I+13:PRINT USING"########";X#(7,I)
```

```
2600 LOCATE 30,I+13:PRINT USING"########";X#(8,I)
2610 NEXT I
2620 LOCATE 3 ,19:PRINT USING"########";A#(5,P)
2630 LOCATE 12,19:PRINT USING"########";A#(6,P)
2640 LOCATE 21,19:PRINT USING"########";A#(7,P)
2650 LOCATE 30,19:PRINT USING"########";A#(8,P)
2660 LOCATE 5,22
2670 PRINT"デ゛ータノ シュウセイヲ シマスカ (y/n)";
2680 I$=INPUT$(1)
2690 IF I$="y" OR I$="Y" THEN 2770
2700 IF I$="n" OR I$="N" THEN 2720
2710 GOTO 2680                                  メッセージ出力&キー入力
2720 LOCATE 5,22
2730 PRINT" ツギ゛ノ Bニ ウツリマスカ (y/n)   ";
2740 I$=INPUT$(1)
2750 IF I$="n" OR I$="N" THEN 70 ELSE P=P+1
2760 IF P=32 THEN 70 ELSE 2110
2770 LOCATE 0,21
2780 PRINT" スウチヲ ウマク カーソルキーヲ ツカッテ ニュウリョクシテ クダ゛サイ。 ";
2790 PRINT"  ニュウリョクガ゛ スミマシタラ RETURNキーヲ オシテクダ゛サイ。";
2800 LOCATE 3,5
2810 LINEINPUT I$
2820 KEY 0,CHR$(30)
2830 FOR I=1 TO 4
2840 LOCATE 0,I+5:LINEINPUT I$(I)          画面読み取りに使用
2850 KEY 0,CHR$(30)+CHR$(13)               X1に↑CRを押させる
2860 NEXT I
2870 FOR I=1 TO 4
2880 LOCATE 0,I+14:LINEINPUT I$(I+4)
2890 KEY 0,CHR$(30)+CHR$(13)
2900 NEXT I
2910 FOR I=1 TO 4
2920 X#(1,I)=VAL(MID$(I$(I),4 ,8))
2930 X#(2,I)=VAL(MID$(I$(I),13,8))
2940 X#(3,I)=VAL(MID$(I$(I),22,8))
2950 X#(4,I)=VAL(MID$(I$(I),31,8))          読み込んだデータを数値に直す
2960 NEXT I
2970 FOR I=1 TO 4
2980 X#(5,I)=VAL(MID$(I$(I+4),4 ,8))
2990 X#(6,I)=VAL(MID$(I$(I+4),13,8))
3000 X#(7,I)=VAL(MID$(I$(I+4),22,8))
3010 X#(8,I)=VAL(MID$(I$(I+4),31,8))
3020 NEXTI
3030 FOR I=1 TO 8:A#(I,P)=0
3040 NEXT I
3050 FOR I=1 TO 8                           変数を修正する
3060 FOR J=1 TO 4
3070 A#(I,P)=A#(I,P)+X#(I,J)
3080 NEXT J:NEXT I
3090 PAUSE 10
3100 GOTO 2190
3110 REM ■■■ グ゛ラフノ シュツリョク ■■■
3120 FOR I=0 TO 8:B#(I)=0:NEXT I:C#=0
3130 CONSOLE 0,25
3140 WIDTH 80:CLS 4:PALET
3150 LOCATE 20,1
3160 PRINT " ■■■ SHARP-X1 Computer カケイボ゛ V.1 ■■■";
3170 LOCATE 20,23
3180 PRINT " ■■ (C)Copyright 1983 By HEART SOFT ■■";
3190 LINE(30,20)-(149*4,89*2),PSET,7,B ―――― 枠を書く
3200 R=30
3210 FOR I=1 TO 8
3220 FOR J=1 TO 31
3230 B#(I)=B#(I)+A#(I,J)                    数値を集計する
3240 NEXT J:NEXT I
3250 FOR I=1 TO 7
3260 C#=C#+B#(I)
3270 NEXT I
3280 IF C#=0 OR B#(8)=0 THEN LOCATE 30,12:PRINT"■■■ デ゛ータガ゛ アリマセン ■■■":FOR I=1 TO
 1000:NEXT I:GOTO 70 ―――― 入金・出金の合計いずれかが0の場合
3290 FOR I=1 TO 7
3300 R(I)=B#(I)/C#*100 ―――― 百分率を計算
3310 NEXT I
```

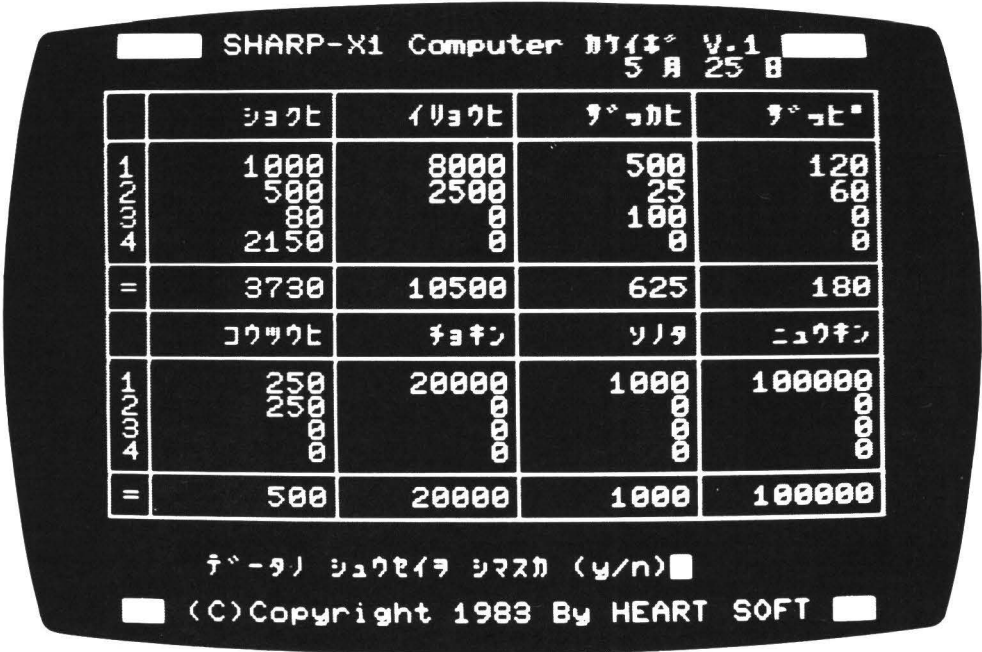
16. コンピューター家計簿

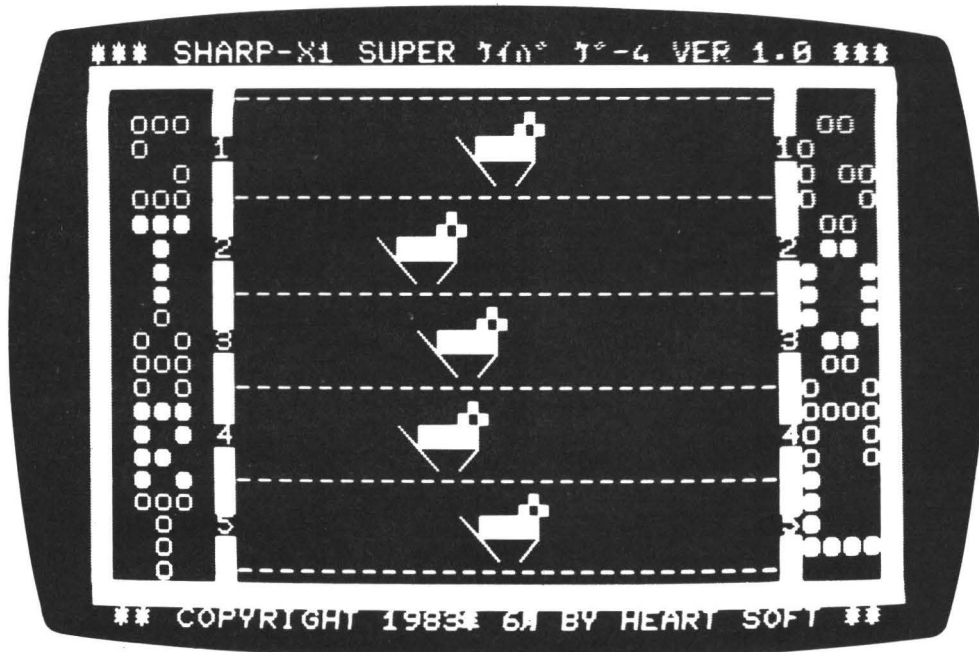
```
3320 FOR I=1 TO 7
3330 LINE (R*4,25*2)-(R*4+R(I)*4,54*2),PSET,I,BF ── 出金の状態を出力
3340 R=R+R(I)
3350 NEXT I
3360 IF C#>B#(8) THEN 3400
3370 LINE(30*4,120)-(130*4,64*2),PSET,6,BF
3380 LINE(30*4,140)-(30*4+(C#/B#(8)*100)*4,74*2),PSET,2,BF    } 出金・入金の比較したグラフ
3390 GOTO 3420
3400 LINE(30*4,120)-(30*4+(B#(8)/C#*100)*4,64*2),PSET,6,BF
3410 LINE(30*4,140)-(130*4,74*2),PSET,2,BF
3420 LOCATE 6,4:PRINT"ｼｼｭﾂｳﾁﾜｹ ";
3430 FOR I=1 TO 7:COLOR I
3440 PRINT K$(I);"    ";
3450 NEXT I
3460 LOCATE 15,5                                              } 項目別の百分率を出力
3470 FOR I=1 TO 7:COLOR I
3480 PRINT INT(R(I));"%    ";
3490 NEXT I
3500 LOCATE 6,14:PRINT"ｾﾞﾝｼｭｳﾆｭｳ";
3510 LOCATE 6,16:PRINT"ｾﾞﾝｼｼｭﾂ";
3520 LOCATE 25,21
3530 PRINT "ﾖﾐｵﾜｯﾀﾗ SPACEｷｰｦ ｵｼﾃ ｸﾀﾞｻｲ｡";                } メッセージ出力&キー入力
3540 I$=INPUT$(1)
3550 IF I$<>" " THEN 3540
3560 GOTO 70
3570 REM ■■■ ｵｼﾏｲ ■■■
3580 PRINT CHR$(12);
3590 LINE(0,0)-(39,24),"X",B
3600 LOCATE 13,10
3610 PRINT"ｺﾞｸﾛｳｻﾏ ﾃﾞｼﾀ｡"
3620 LOCATE 10,14
3630 PRINT"■■■ By HEART SOFT ■■■"
3640 END
```

SHARP-X1 Computer ｶｹｲｷﾞ V.1
5 月 25 日

	ｼｮｸﾋ	ｲﾘｮｳﾋ	ｻﾞｯｶﾋ	ｻﾞｯﾋﾟ
1	1000	8000	500	120
2	500	2500	25	60
3	80	0	100	0
4	2150	0	0	0
=	3730	10500	625	180
	ｺｳﾂｳﾋ	ﾁｮｷﾝ	ｿﾉﾀ	ﾆｭｳｷﾝ
1	250	20000	1000	100000
2	250	0	0	0
3	0	0	0	0
4	0	0	0	0
=	500	20000	1000	100000

ﾃﾞｰﾀﾉ ｼｭｳｾｲｦ ｼﾏｽｶ (y/n)

(C)Copyright 1983 By HEART SOFT

コンピューター家計簿のデータ入力画面

17
競馬ゲーム

日本では賭けごとは禁止されていますが，唯一認められているのが競馬，競輪，ボート等です。その中でも一番人気の高い競馬ゲームを作ってみました。このゲームは1～5枠まであり，単勝式で，複勝ではありません。ここでは完全にランダムですので，人気馬，穴馬等はありませんが，配当倍率もランダムですので，せっかく当てても倍率が1倍ということもあります。なお，倍率は1～10倍の間になります。

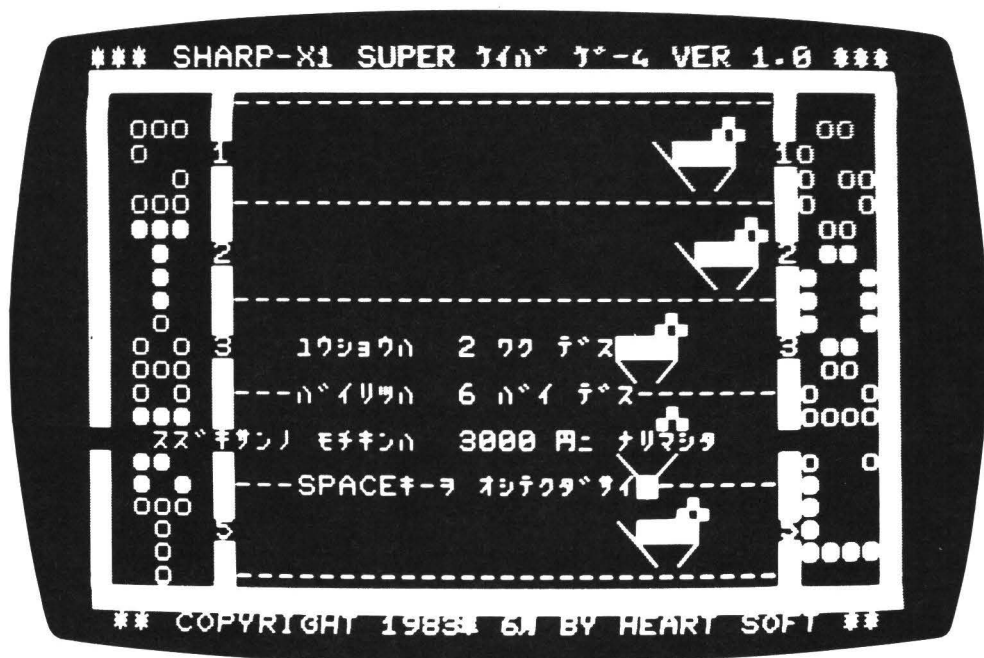
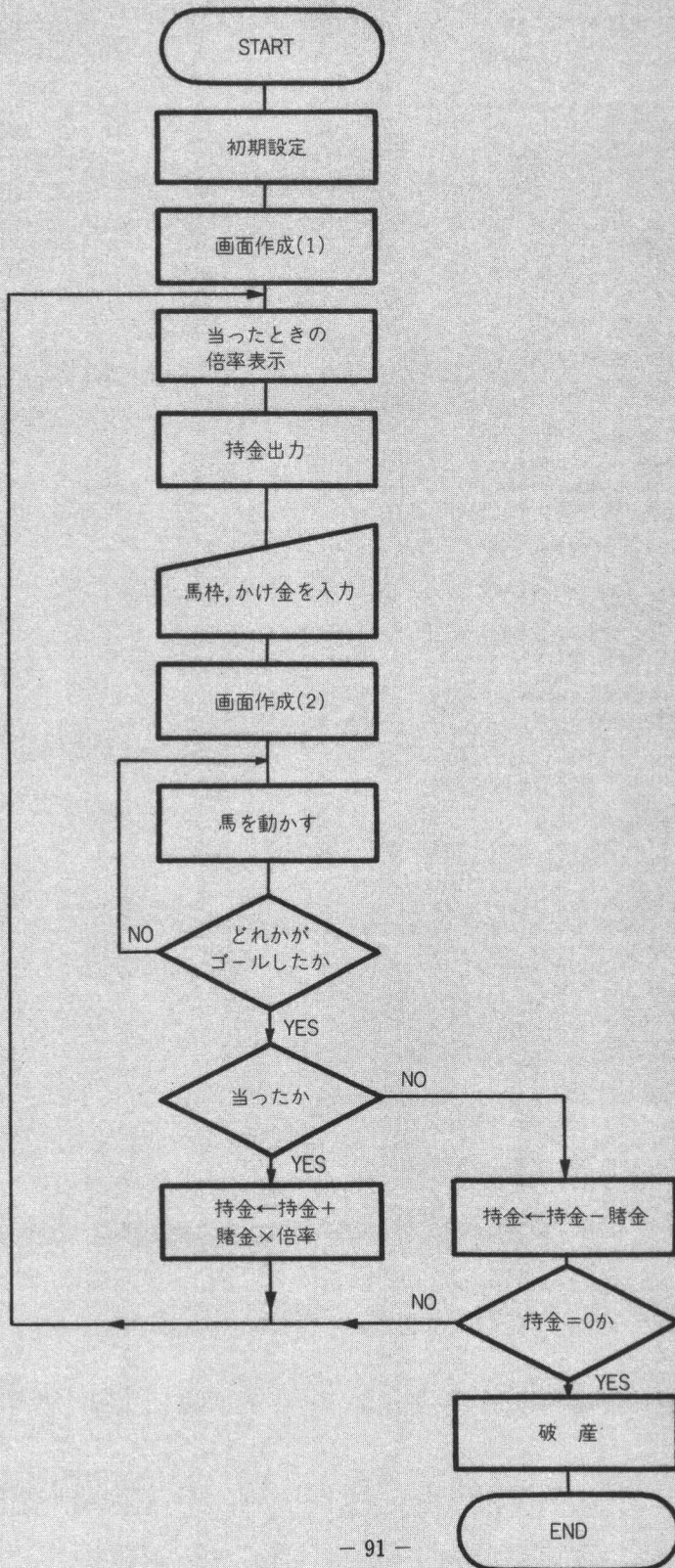
最初，馬を動かすルーチンを作ったとき，I=INT(RND(1)*6):ON I GOTO—としていたのですが，X1のRND関数は数の少ない方が圧倒的に多く出て，1～2枠が勝ってばかりいました。これではゲームになりませんので，各馬を動かすルーチンを作り，そこで馬を動かすか動かさないか$\frac{1}{2}$の確率を使い馬を動かすことにしました。ただ，これでも1，2，3…5の順に馬を動かしているため，数の少ない方が有利ともいえるのですが，実際のゲームはほとんど影響がありませんので御安心を!!

ゲームの遊び方

プログラムをＲＵＮすると，まず何人で遊ぶか聞いてきます。最高は９人までで，各自に名前を入力して下さい。各馬の名前が発表され，どの馬にいくら賭けるかを入力します。すると画面が変わり，ファンファーレと共に馬が走り出します。後は自分が賭けた馬が勝つことを祈るだけです。勝った場合は賭金×倍率が得点になります。

変数表

Ｕ＄(１)～Ｕ＄(６)→馬のデータ
Ｓ(ｎ)→持ち金
Ｓ＄(ｎ)→プレイヤーの名前
Ｘ１～Ｘ５→各馬のＹ座標
Ｗ(ｎ)→買う枠
Ｋ(ｎ)→買う枚数
ＷＩＮ→勝った馬番
Ｑ→倍率
Ｈ，Ｉ，Ｊ，Ｋ→ループ用

競馬ゲーム・フローチャート
START
初期設定
画面作成(1)
当ったときの
倍率表示
持金出力
馬枠,かけ金を入力
画面作成(2)
馬を動かす
NO
どれかが
ゴールしたか
YES
当ったか
NO
YES
持金←持金+
賭金×倍率
持金←持金−賭金
NO
持金=0か
YES
破　産
END

17. 競馬ゲーム

```
1 REM*************************************
2 REM*                                   *
3 REM* ■■■ SUPER ｹｲﾊﾞｹﾞｰﾑ VER 1.0 ■■■ *
4 REM*                                   *
5 REM*      [C]COPYRIGHT 1983年 6月      *
6 REM*                                   *
7 REM*  FOR SHARP-X1 BY HEART SOFT       *
8 REM*                                   *
9 REM*************************************
10 WIDTH 40:CONSOLE 0,25
20 COLOR 7,0:CLS
30 U$(1)="     ▟▙ "
40 U$(2)="  ／■■ "
50 U$(3)="  ／＼  "
60 U$(4)="     ▟▙ "
70 U$(5)="  ＼■■ "
80 U$(6)="  ＼／  "
90 LOCATE 10,15
100 INPUT"ﾅﾝﾆﾝﾃﾞ ｱｿﾋﾞﾏｽｶ";M
110 FOR K=1 TO M
120 LOCATE 0,17:PRINT SPACE$(39);
130 LOCATE 8,17
140 PRINT K;:INPUT"ﾊﾞﾝﾉ ｵﾅﾏｴﾊ ";S$(K)
150 S(K)=2000:NEXT K
160 COLOR 7,0
170 X1=7:X2=7:X3=7:X4=7:X5=7:CLS
180 LOCATE 0,0:PRINT"■■■ SHARP-X1 SUPER ｹｲﾊﾞ ｹﾞｰﾑ VER 1.0 ■■■";
190 LOCATE 0,24:PRINT"■■■ COPYRIGHT 1983.6. BY HEART SOFT ■■■";
200 LOCATE 0,8
210 PRINT"    1ﾜｸ....ﾊｲｾｲｺｰ    2ﾜｸ....ﾀｹﾎｰﾌﾟ"
220 PRINT
230 PRINT"    3ﾜｸ....ｱｽﾞﾏﾊﾝﾀｰ  4ﾜｸ....ｷﾀﾉｶﾁﾄﾞｷ"
240 PRINT
250 PRINT"    5ﾜｸ....ﾄｳｼｮｳ ﾎﾞｰｲ"
260 LINE(0,6)-(39,14),"■",B
270 FOR K=1 TO M
280 LOCATE 0,18:PRINT SPACE$(39);
290 IF S(K)=0 THEN 410
300 LOCATE 5,18
310 PRINT S$(K);"ｻﾝﾉ ﾓﾁｷﾝﾊ ";S(K);"円 ﾃﾞｽ";
320 LOCATE 0,20:PRINT SPACE$(39);
330 LOCATE 12,20
340 INPUT"ﾅﾝﾜｸｦ ｶｲﾏｽｶ";W$(K)
350 IF VAL(W$(K))<1 OR VAL(W$(K))>5 THEN 320
360 LOCATE 0,20:PRINT SPACE$(39);
370 LOCATE 8,20
380 INPUT"ﾅﾝﾏｲ ｶｲﾏｽｶ (1ﾏｲ 100円)";K$(K)
390 IF VAL(K$(K))<1 OR VAL(K$(K))*100>S(K) THEN 360
400 W(K)=VAL(W$(K)):K(K)=VAL(K$(K))
410 NEXT K
420 GOSUB 1620:H=0
430 FOR I=1 TO 5
440 FOR J=1 TO 3
450 LOCATE 7,I*3+J+H-1
460 PRINT U$(J);
470 NEXT J:H=H+1:NEXT I:H=1
480 GOSUB 1920
490 GOSUB 2050
500 ON H GOTO 530,680,830,980,1130
510 H=H+1:IF H=6 THEN H=1
520 GOTO 490
530 REM NO.1
540 IF INT(RND(1)*2)<>0 THEN 510
550 X1=X1+1
560 IF X1=29 THEN WIN=1:GOTO 1280
570 FOR I=1 TO 3
580 FOR J=1 TO 3
590 LOCATE X1,2+J
600 PRINT U$(J);
610 NEXT J
620 FOR J=1 TO 3
630 LOCATE X1,2+J
640 PRINT U$(J+3);
650 NEXT J
```

（30～80）馬のデータ
自分の好きなように変えてもいい

（100～140）人数，氏名の入力

（210～250）枠と馬名の出力
ここも好きなように！

（280～310）持ち金の出力，無ければパスされる

（320～350）買う馬券の枠を入力する

（360～400）馬券の枚数を入力する
枚数×100円が持ち金を
オーバーするとやり直し

（420～470）スタート前の馬を表示する

（480～490）ファンファーレとスタート音

（500～520）各馬を動かすルーチンへ行く
どれかゴールするまでくり返す

（560）ゴールしたかの判定X＝29がゴール

（530～650）第1コース

（590～640）馬を動かすルーチン

```
660 NEXT I
670 GOTO 510
680 REM NO.2
690 IF INT(RND(1)*2)<>0 THEN 510
700 X2=X2+1
710 IF X2=29 THEN WIN=2:GOTO 1280
720 FOR I=1 TO 3
730 FOR J=1 TO 3
740 LOCATE X2,6+J                       第2コース
750 PRINT U$(J);
760 NEXT J
770 FOR J=1 TO 3
780 LOCATE X2,6+J
790 PRINT U$(J+3);
800 NEXT J
810 NEXT I
820 GOTO 510
830 REM NO.3
840 IF INT(RND(1)*2)<>0 THEN 510
850 X3=X3+1
860 IF X3=29 THEN WIN=3:GOTO 1280
870 FOR I=1 TO 3
880 FOR J=1 TO 3
890 LOCATE X3,10+J                      第3コース
900 PRINT U$(J);
910 NEXT J
920 FOR J=1 TO 3
930 LOCATE X3,10+J
940 PRINT U$(J+3);
950 NEXT J
960 NEXT I
970 GOTO 510
980 REM NO.4
990 IF INT(RND(1)*2)<>0 THEN 510
1000 X4=X4+1
1010 IF X4=29 THEN WIN=4:GOTO 1280
1020 FOR I=1 TO 3
1030 FOR J=1 TO 3
1040 LOCATE X4,14+J
1050 PRINT U$(J);                       第4コース
1060 NEXT J
1070 FOR J=1 TO 3
1080 LOCATE X4,14+J
1090 PRINT U$(J+3);
1100 NEXT J
1110 NEXT I
1120 GOTO 510
1130 REM NO.5
1140 IF INT(RND(1)*2)<>0 THEN 510
1150 X5=X5+1
1160 IF X5=29 THEN WIN=5:GOTO 1280
1170 FOR I=1 TO 3
1180 FOR J=1 TO 3
1190 LOCATE X5,18+J
1200 PRINT U$(J);                       第5コース
1210 NEXT J
1220 FOR J=1 TO 3
1230 LOCATE X5,18+J
1240 PRINT U$(J+3);
1250 NEXT J
1260 NEXT I
1270 GOTO 510
1280 COLOR 7
1290 GOSUB1950               ―― ゴールの音
1300 LOCATE 10,12
1310 PRINT"ﾕｳｼｮｳﾊ ";WIN;"ﾜｸ ﾃﾞｽ";              レースの優賞，倍率の発表
1320 Q=INT(RND(1)*10)+1      ―― 倍率を1～10までに決める
1330 LOCATE 10,14
1340 PRINT"ﾊﾞｲﾘﾂﾊ ";Q;"ﾊﾞｲ ﾃﾞｽ";
1350 FOR K=1 TO M
1360 IF S(K)=0 THEN 1430
1370 S(K)=S(K)-K(K)*100:IF W(K)=WIN THEN S(K)=S(K)+Q*K(K)*100 ―― 持金の計算，持金から賭金を引き，当った金をたす
1380 IF S(K)=0 THEN 1480     ―― 破産したとき
1390 LOCATE 0,16:PRINT SPACE$(39);
```

17. 競馬ゲーム

```
1400 LOCATE 3,16
1410 PRINT S$(K);"ｻﾝﾉ ﾓﾁｷﾝﾊ ";S(K);"円ﾆ ﾅﾘﾏｼﾀ";
1420 PAUSE 30
1430 NEXT K
1440 LOCATE 10,18
1450 PRINT"SPACEｷｰｦ ｵｼﾃｸﾀﾞｻｲ";
1460 I$=INPUT$(1):IF I$<>" " THEN 1460
1470 GOTO 160
1480 B=0:FOR T=1 TO M
1490 B=B+S(T):NEXT
1500 IF B=0 THEN 1540
1510 LOCATE 10,16
1520 PRINT S$(K);"ｻﾝ ﾊ ﾊｻﾝｼﾃ ｼﾏｲﾏｼﾀ";
1530 GOTO 1420
1540 LOCATE 7,16
1550 PRINT "ﾐﾅｻﾝ ｾﾞﾝｲﾝ ﾊｻﾝｼﾃ ｼﾏｲﾏｼﾀ";
1560 LOCATE 8,18
1570 PRINT"ﾓｳ 1ﾄﾞ ｱｿﾋﾞﾏｽｶ? (Y/N)";
1580 I$=INPUT$(1)
1590 IF I$="Y" OR I$="y" THEN RUN
1600 IF I$="N" OR I$="n" THEN CLS:END
1610 GOTO 1580
1620 REM
1630 CLS:COLOR 7,1
1640 PRINT"*** SHARP-X1 SUPER ｹｲﾊﾞ ｹﾞｰﾑ VER 1.0 ***";
1650 LOCATE 0,24:PRINT" ** COPYRIGHT 1983年 6月 BY HEART SOFT **";
1660 COLOR 6
1670 LOCATE 0,1
1680 PRINT"■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■";
1690 PRINT"■     ■----------------------------■    ■";
1700 PRINT"■ ooo ■                            ■ oo ■";
1710 PRINT"■ o   1                            1o   ■";
1720 PRINT"■   o ■                            ■o oo■";
1730 PRINT"■ ooo ■----------------------------■o  o■";
1740 PRINT"■ ●●● ■                            ■ oo ■";
1750 PRINT"■  ●  2                            2 ●● ■";
1760 PRINT"■  ●  ■                            ■●  ●■";
1770 PRINT"■  ●  ■----------------------------■●  ●■";
1780 PRINT"■  o  ■                            ■●  ●■";
1790 PRINT"■ o o 3                            3 ●● ■";
1800 PRINT"■ ooo ■                            ■ oo ■";
1810 PRINT"■ o o ■----------------------------■o  o■";
1820 PRINT"■ ●●● ■                            ■oooo■";
1830 PRINT"■ ● ● 4                            4o  o■";
1840 PRINT"■ ●●  ■                            ■o  o■";
1850 PRINT"■ ● ● ■----------------------------■●   ■";
1860 PRINT"■ ooo ■                            ■●   ■";
1870 PRINT"■  o  5                            5●   ■";
1880 PRINT"■  o  ■                            ■●●●●■";
1890 PRINT"■  o  ■----------------------------■    ■";
1900 PRINT"■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■";
1910 COLOR 4:RETURN
1920 TEMPO 500:PLAY "V15"
1930 PLAY "O4FAO5C9O4A7F7A7F7C9R4
1940 PLAY "F7A7O5C9O4A7F7C7F8R9
1950 SOUND 1,3   :SOUND 0,250
1960 SOUND 3,3   :SOUND 2,250
1970 SOUND 5,30  :SOUND 4,250
1980 SOUND 6,10
1990 SOUND 7,&HE8
2000 SOUND 8,16  :SOUND 9,16
2010 SOUND 10,16
2020 SOUND 12,18:SOUND 11,20
2030 SOUND 13,0 :FOR I=0 TO 900:NEXT
2040 RETURN
2050 SOUND 1,30  :SOUND 0,250
2060 SOUND 3,30  :SOUND 2,250
2070 SOUND 5,30  :SOUND 4,250
2080 SOUND 6,20
2090 SOUND 7,&HF8
2100 SOUND 8,16  :SOUND 9,16
2110 SOUND 10,16
2120 SOUND 12,2 :SOUND 11,20
2130 SOUND 13,0 :FOR I=0 TO 90:NEXT
2140 SOUND 13,0 :FOR I=0 TO 90:NEXT
2150 RETURN
```

持金の表示（1400〜1410）

全員とも破産したとき（1500）

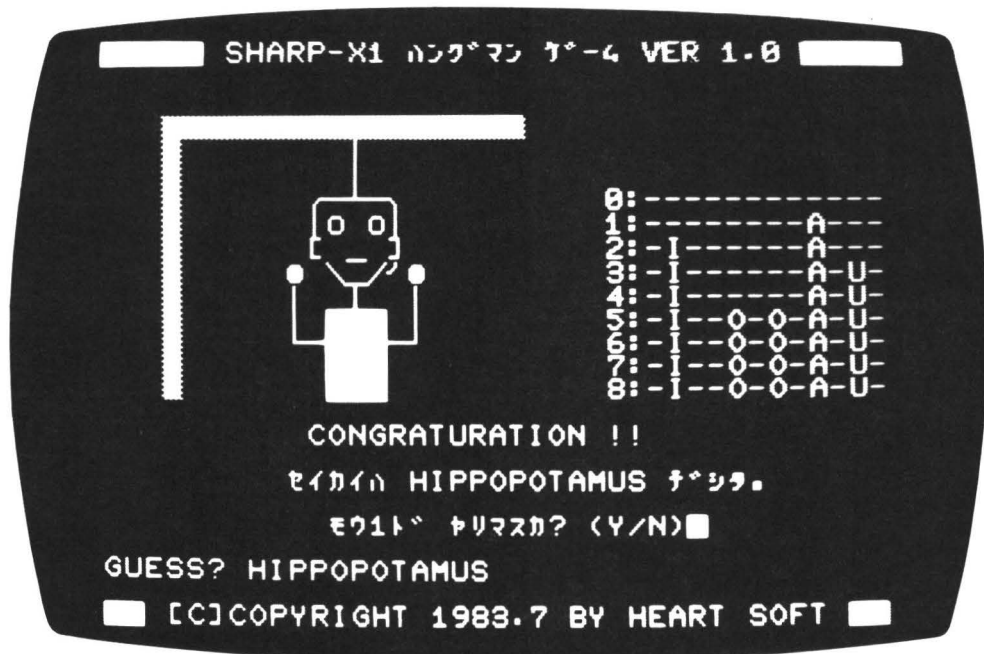
メッセージ出力（1510〜1550）

再ゲームか？（1560〜1610）

破産したときのルーチン（1480〜1610）

画面表示（1640〜1910）

以下音のサブルーチン（1920〜）

18
ハングマン

あなたの友達が無実の罪で絞首刑になろうとしています。友達を助けるためには，あなたが，X１の出した単語を当てなくてはいけません。さあ，あなたは友達を助けることが，できるでしょうか？（何故，単語を当てるだけで友人が絞首刑を免がれるかは，あまり深く考えないで下さい。）

単語は全部で 30 個用意しましたが，いくらでも増やすことができます。その方法は以下の通りです。

50行　ＤＩＭ　Ｗ＄(30) →ＤＩＭ　Ｗ＄（増やした単語数）

50行　Ｗ＝31 →Ｗ＝増やした単語数＋１

10050 行から DATA 文で単語を入れる

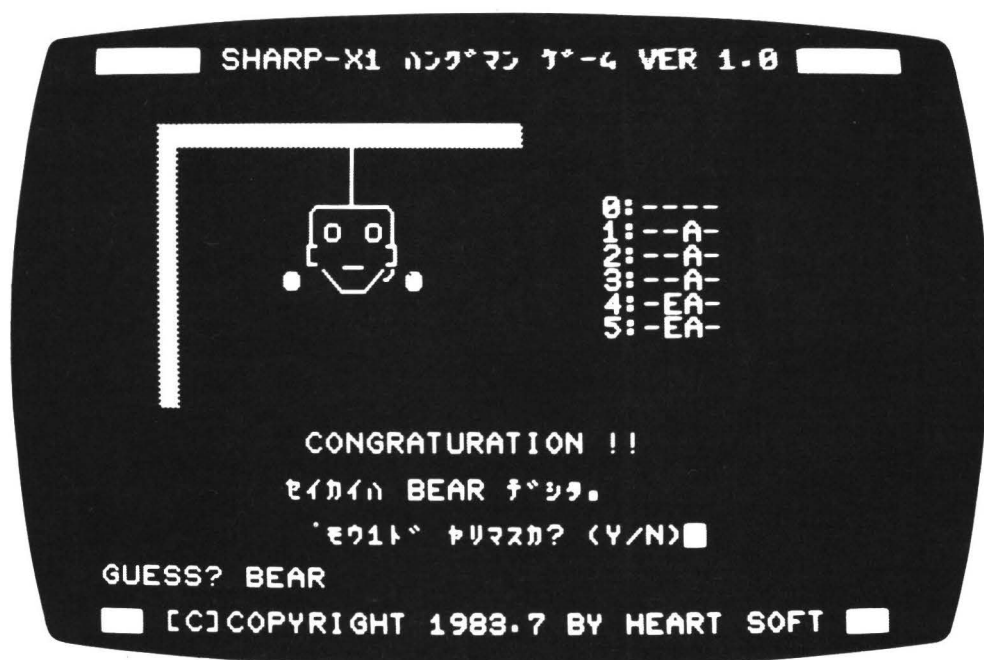
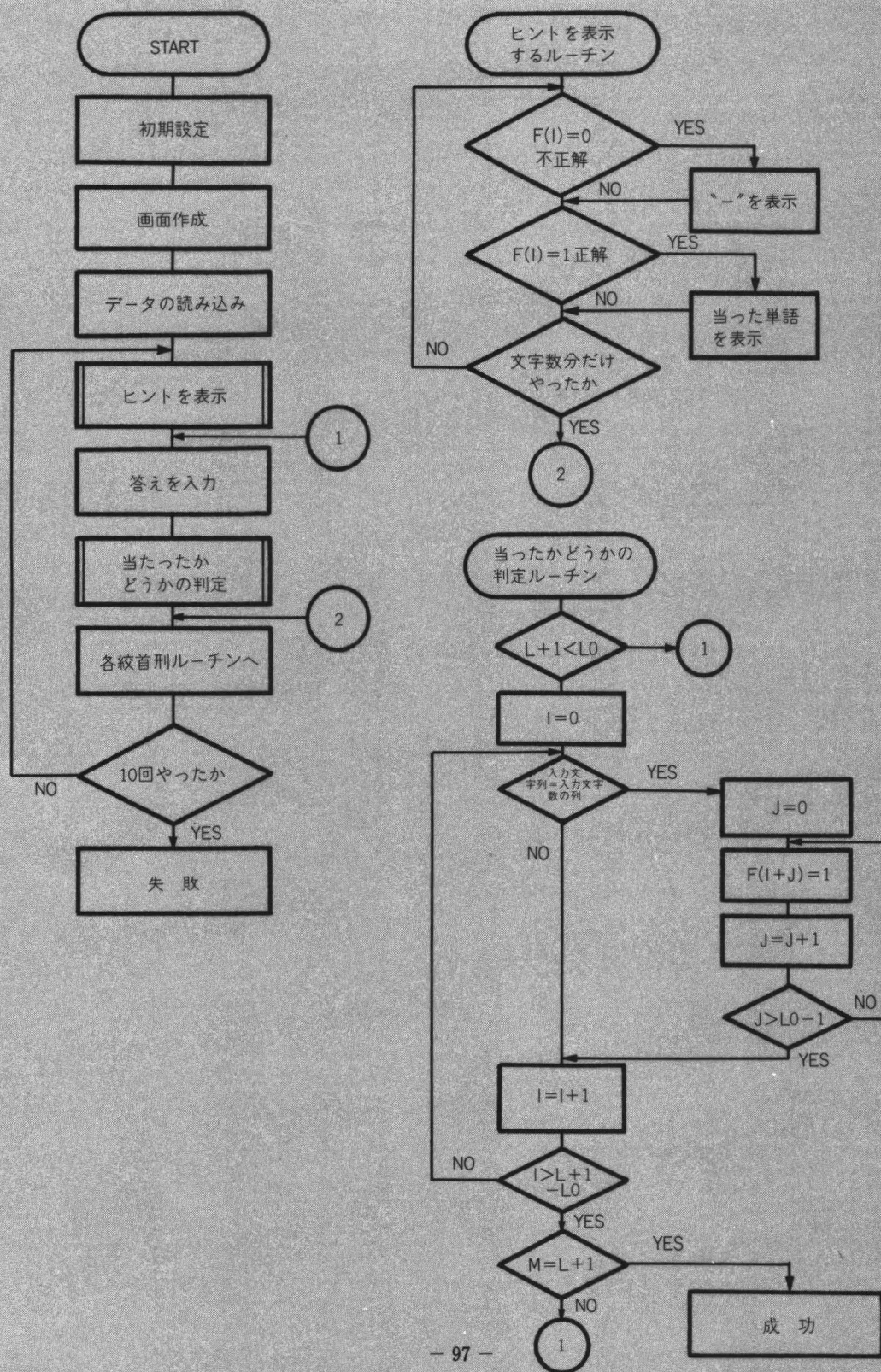
ゲームの遊び方

Ｘ１の出した単語を当てるのですが，チャンスは10回です。答えは，アルファベット１文字ずつ入力して行き，あなたが間違うたびに絞首刑が進んでいきます。答えが分ったときは，フルスペルで入力してもかまいません。

このゲームのコツですが，まず，ア，イ，ウ，エ，オ（Ａ，Ｉ，Ｕ，Ｅ，Ｏ）の母音を入力してみます。そうすればある程度は分ると思います。イカサマとしては，リストをプリンターで出しておくという手もありますが………。

変数表

Ｗ→単語の数
Ｐ→ランダムに選んだ単語の番号
Ｌ→単語の文字数
Ｍ→当ったアルファベット
Ｙ→ヒントを出すＹ座標
Ｆ(ｎ)→当った部分
Ｇ＄→入力した文字列
Ｌ０→入力した文字数
Ｉ＄→キー入力（再ゲーム？）
Ｇ，Ｉ→ループ用

ハングマン・フローチャート
START
初期設定
画面作成
データの読み込み
ヒントを表示
1
答えを入力
当たったか
どうかの判定
2
各絞首刑ルーチンへ
10回やったか
NO
YES
失　敗
ヒントを表示
するルーチン
F(I)=0
不正解
YES
NO
"−"を表示
F(I)=1正解
YES
NO
当った単語
を表示
NO
文字数分だけ
やったか
YES
2
当ったかどうかの
判定ルーチン
L+1<L0
1
I=0
入力文
字列=入力文字
数の列
YES
J=0
F(I+J)=1
J=J+1
NO
J>L0−1
YES
NO
I=I+1
NO
I>L+1
−L0
YES
M=L+1
YES
成　功
NO
1

18. ハングマン

```
1 REM**************************************
2 REM*                                    *
3 REM*  ■■■■ ﾊﾝｸﾞﾏﾝ ｹﾞｰﾑ VER 1.0 ■■■■  *
4 REM*                                    *
5 REM*     [C]Copyright 1983年 7月        *
6 REM*                                    *
7 REM*   FOR SHARP-X1 By Heart Soft       *
8 REM*                                    *
9 REM**************************************
10 CONSOLE 0,25:WIDTH 40                          ┐
20 CLS:CLICK OFF:COLOR 7,0                        │
30 RANDOMIZE:TIME=0                               │ 初期設定
40 CLS:LOCATE 10,15                               │
50 DIM W$(30),F(14):W=31                          │
90 CLS:COLOR 7:LOCATE 0,0                         ┘
100 PRINT"■■■■ SHARP-X1 ﾊﾝｸﾞﾏﾝ ｹﾞｰﾑ VER 1.0 ■■■■";  ┐
110 LOCATE 0,24                                   │ 画面作成
120 PRINT"■■ [C]COPYRIGHT 1983.7 BY HEART SOFT ■■";  │
130 FOR I=0 TO 30:READ W$(I):NEXT I               ┘
250 P=INT(RND(1)*W)              ――――― データの読み込み
260 L=LEN(W$(P))-1
270 FOR I=0 TO L:F(I)=0:NEXT I
280 FOR G=0 TO 9
290 GOSUB 1000                   ――――― ヒントを表示
300 LOCATE 0,22:INPUT"GUESS";G$  ――――― キー入力
310 GOSUB 2000                   ――――― 当ったかどうかの判定
330 IF M=L+1 THEN 500
340 GOSUB 3000                   ――――― 各処理ルーチンへ
350 NEXT G
360 G=10:GOSUB 1000
370 LINE(0,15)-(39,21)," ",BF                     ┐
380 LOCATE 0,16                                   │
390 PRINT"             GAME OVER !!               │
400 PRINT                                         │
410 PRINT"        ｾｲｶｲﾊ ";W$(P);" ﾃﾞｼﾀ.           │ 失敗したとき
420 PRINT                                         │
430 PRINT"          ﾓｳ1ﾄﾞ ﾔﾘﾏｽｶ? (Y/N)";          │
440 I$=INPUT$(1)                                  │
450 IF I$="Y" OR I$="y" THEN RUN                  │
460 IF I$="N" OR I$="n" THEN CLS:END              │
470 GOTO 440                                      ┘
500 REM
505 LINE(0,15)-(39,21)," ",BF                     ┐
510 LOCATE 0,16                                   │
520 PRINT"          CONGRATURATION !!             │
530 PRINT                                         │
540 PRINT"        ｾｲｶｲﾊ ";W$(P);" ﾃﾞｼﾀ.           │ 成功したとき
550 PRINT                                         │
560 PRINT"          ﾓｳ1ﾄﾞ ﾔﾘﾏｽｶ? (Y/N)";          │
570 I$=INPUT$(1)                                  │
580 IF I$="Y" OR I$="y" THEN RUN                  │
590 IF I$="N" OR I$="n" THEN CLS:END              │
600 GOTO 570                                      ┘
1000 REM
1005 Y=G+6:LOCATE 24,Y                            ┐
1010 IF G=10 THEN LOCATE 23,Y                     │
1020 PRINT G                                      │
1030 LOCATE 26,Y:PRINT":";                        │ 当ったところの単語を表示する
1050 FOR I=0 TO L                                 │
1060 IF F(I)=0 THEN PRINT"-";                     │
1070 IF F(I)=1 THEN PRINT MID$(W$(P),I+1,1);      │
1080 NEXT I                                       ┘
1100 RETURN
2000 REM
2005 L0=LEN(G$):IF L0=0 THEN M=0:RETURN                                 ┐
2010 IF L+1<L0 THEN RETURN                                              │
2020 FOR I=0 TO L+1-L0                                                  │
2030 IF G$=MID$(W$(P),I+1,L0) THEN FOR J=0 TO L0-1:F(I+J)=1:NEXT J     │
2040 NEXT I                                                             │ 正解したか
2050 M=0                                                                │ どうかの判定
2060 FOR I=0 TO L                                                       │
2070 IF F(I)=1 THEN M=M+1                                               │
2080 NEXT I                                                             │
2090 RETURN                                                             ┘
```

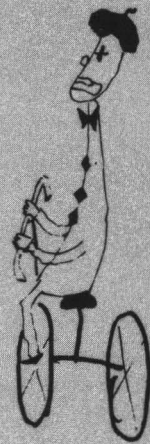
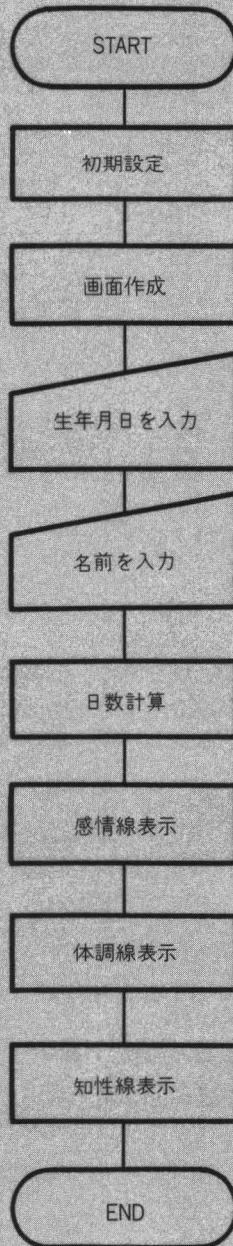
```
3000 REM
3001 TEMPO 500:PLAY"O3C:O3E"
3010 ON G+1 GOSUB "1","2","3","4","5","6","7","8","9","10"  } 各表示ルーチンへ行く
3020 RETURN
3030 LABEL "1"                                   ┐
3040 LINE(3,3)-(20,3),"■"                        } No. 1
3050 RETURN                                      ┘
3060 LABEL "2"                                   ┐
3070 LINE(3,3)-(3,20),"■"                        } No. 2
3080 RETURN                                      ┘
3090 LABEL "3"                                   ┐
3100 LOCATE 8, 4:PRINT"     |                    } No. 3
3110 LOCATE 8, 5:PRINT"     |                    
3120 RETURN                                      ┘
3130 LABEL "4"                                   ┐
3140 LOCATE 8, 6:PRINT"   ┌─┴─┐                  } No. 4
3150 LOCATE 8, 7:PRINT"   |o o|                  
3160 RETURN                                      ┘
3170 LABEL "5"                                   ┐
3180 LOCATE 8, 8:PRINT"   [ _ ]                  } No. 5
3190 LOCATE 8, 9:PRINT" ● ╲_╱ ●                  
3200 RETURN                                      ┘
3210 LABEL "6"                                   ┐
3220 LOCATE 8,10:PRINT" |  |  |                  } No. 6
3230 LOCATE 8,11:PRINT" | ■■ |                   
3240 LOCATE 8,12:PRINT" └─■■─┘                   
3250 RETURN                                      ┘
3260 LABEL "7"                                   ┐
3270 LOCATE 8,13:PRINT"   ■■                     } No. 7
3280 LOCATE 8,14:PRINT"   ■■                     
3290 RETURN                                      ┘
3300 LABEL "8"                                   ┐
3310 LOCATE 8,15:PRINT"   ■■                     } No. 8
3320 LOCATE 8,16:PRINT"   ╱╲                     
3330 RETURN                                      ┘
3340 LABEL "9"                                   ┐
3350 LOCATE 8,17:PRINT"  ╱   ╲                   } No. 9
3360 LOCATE 8,18:PRINT" ╱     ╲                  
3370 LOCATE 8,19:PRINT"●       ●                 
3375 RETURN                                      ┘
3380 LABEL "10"                                  ┐
3390 PLAY"O5CO4BAGFEDC"                          } No. 10
3400 RETURN                                      
10000 REM                                        ┘
10010 DATA HORSE,PIG,CAMEL,BUFFALO,DONKEY,REINDEER,DOG,CAT,MOUSE                ┐
10020 DATA SQUIRREL,RABBIT,FOX,BAT,WHALE,DOLPHIN,ANTELOPE,KANGAROO              } 単語のデータ
10030 DATA BEAR,SEAL,WOLF,MONKEY,BABOON,GORILLA,GIRAFFE,LION,LEOPARD            
10040 DATA TIGER,HIPPOPOTAMUS,ELEPHANT,ZEBRA,RHINOCEROS                         ┘
```

19
バイオリズム(G-RAM使用)

人間の体調というものはあるリズムを持って変化するといいます。その変化を前もって知っておけば体調の悪いときに無理をしなくても良くなります。その体調をコンピューターで調べようというのが，このバイオリズムです。体調は 23 日，知性は 33 日，感情は 28 日の周期で変化します。ですから生まれてから経った日々を割ったときの余りを調べてその数値によって判断を下します。グラフの見方ですが，中心線を0として上方を高周期，下方を低調期といい，両極端はいいのですが，中心線が重なったときは最悪ですので，その日は外出などはさけて，早く寝てしまいましょう。

プログラムの走らせ方

まず，生まれた年，月，日を入力し，次に調べる年，月を入力して下さい。年は西暦での入力です。5日おきに目盛が打ってありますので，調べる日が大体分ると思います。1分位で表示が終りますので，次の日に移るときは，好きなキーを押して下さい。

バイオリズム・フローチャート
START
初期設定
画面作成
生年月日を入力
名前を入力
日数計算
感情線表示
体調線表示
知性線表示
END

19. バイオリズム

```
1 REM************************************
2 REM*                                  *
3 REM*   ■■■■ BIORHYTHM VER 1.0 ■■■■    *
4 REM*                                  *
5 REM*      [C]Copyright 1983年 7月      *
6 REM*                                  *
7 REM*   FOR SHARP-X1 By Heart Soft     *
8 REM*                                  *
9 REM************************************
10 CONSOLE 0,25:WIDTH 80:CLICK OFF
20 RANDOMIZE:COLOR 7:OPTION SCREEN 1:INIT "CRT:":PALET
30 CLS 4:PRINT"■■■■ SHARP-X1 BIORHYTHM VER 1.0 [C]COPYRIGHT 1983年 7月 MADE BY HE
ART SOFT ■■■■";
40 DIM MN(12,1)
50 DEFFNM(Y)=-(Y MOD 4=0)AND(Y MOD 400>0)
```

関数の定義

```
60 FOR I=0 TO 1:FOR J=1 TO 12
70 READ MN(J,I):NEXT J:NEXT I
```

データの読み込み

```
80 LOCATE 0,2
90 PRINT"           1         5         10         15         20         25         30
100 PRINT" ────┬────────┬────────┬────────┬────────┬────────┬────────┬
────
110 LOCATE 0,12
120 PRINT" ────┼────────┼────────┼────────┼────────┼────────┼────────┼
────
130 LOCATE 0,21
140 PRINT" ────┴────────┴────────┴────────┴────────┴────────┴────────┴
────
```

画面作成

```
150 CONSOLE 22,3
160 LOCATE 5,22:INPUT"ｳﾏﾚﾀ 年ﾊ";BY
170 IF BY<1900 THEN BY=BY+1900
180 LOCATE 25,22:INPUT"ｳﾏﾚﾀ 月ﾊ";BM
190 IF BM<1 OR BM>12 THEN 180
200 LOCATE 45,22:INPUT"ｳﾏﾚﾀ 日ﾊ";BD
210 IF BD<1 OR BD>MN(BM,FNM(BY)) THEN 200
220 LOCATE 5,23:INPUT"ｼﾗﾍﾞﾙ 年ﾊ";NY
230 IF BY<1900 THEN Y=Y+1900
240 LOCATE 25,23:INPUT"ｼﾗﾍﾞﾙ 月ﾊ";NM
250 IF BM<1 OR BM>12 THEN 180
260 M2=FNM(BY):AL=-BD
```

生年月日の入力，調査日の指定

```
270 FOR M=BM TO 12:AL=AL+MN(M,M2):NEXT
280 IF BY=NY THEN 330
290 FOR Y=BY+1 TO NY
300 M2=FNM(Y)
310 AL=AL+365+M2
320 NEXT
330 M2=FNM(NY)
```

日数の計算

```
340 FOR M=NM TO 12:AL=AL-MN(M,M2):NEXT
350 PRINT:PRINT:LOCATE 0,22
360 COLOR 6
370 LOCATE 5,22:PRINT"ｷｲﾛﾉ ｾﾝ=ﾀｲﾁｮｳ"
380 COLOR 2
390 LOCATE 25,22:PRINT"ｱｶﾉ ｾﾝ=ｶﾝｼﾞｮｳ"
400 COLOR 4
410 LOCATE 45,22:PRINT"ﾐﾄﾞﾘﾉ ｾﾝ=ﾁｾｲ"
```

画面表示

```
420 IF AL>23*28*33 THEN AL=AL-23*28*33:GOTO 420
430 P=AL MOD 23:E=AL MOD 28:M=AL MOD 33
440 FOR X=-3 TO 33 STEP .1
450 S=X*16+56:L=(100-28*SIN((X+P)/23*6.28)*2)
460 PSET (S,L,6):IF INT(L)=98 THEN POLY(S,L),6,6,144,90,810
470 S=X*16+56:L=(100-28*SIN((X+E)/28*6.28)*2)
480 PSET (S,L,2):IF INT(L)=98 THEN POLY(S,L),6,2,144,90,810
490 S=X*16+56:L=(100-28*SIN((X+M)/31*6.28)*2)
500 PSET (S,L,4):IF INT(L)=98 THEN POLY(S,L),6,4,144,90,810
```

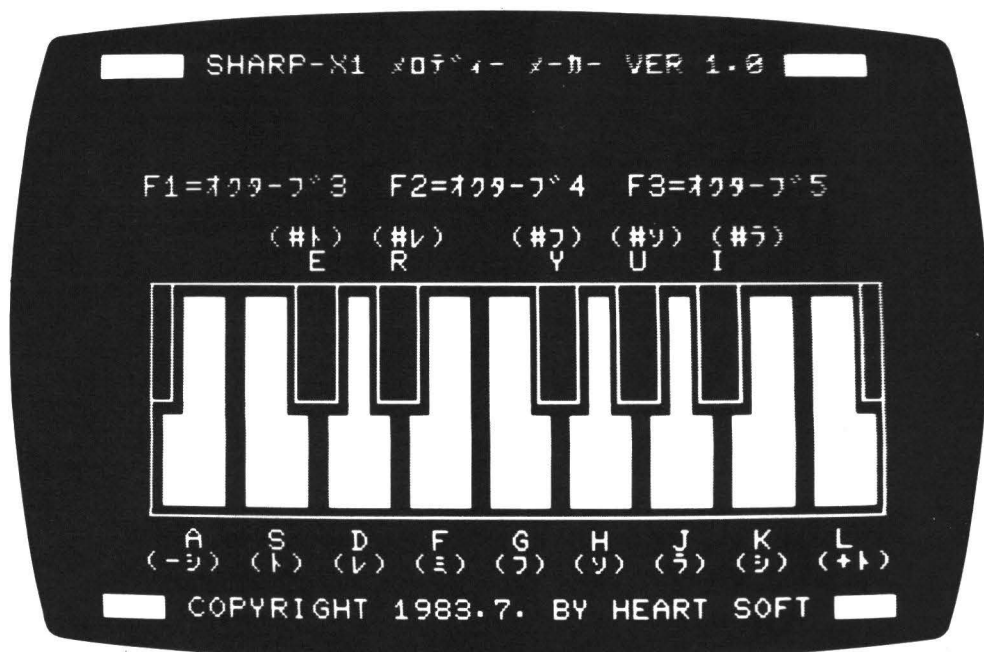
体調線，感情線，知性線をえがく

```
510 NEXT X
520 COLOR 7:CFLASH 1
530 LOCATE 65,22:PRINT"HIT ANY KEY !!"
540 I$=INKEY$:IF I$="" THEN 540
550 RUN
```

キー入力　ANY KEYで再スタート

```
560 DATA 31,28,31,30,31,30,31,31,30,31,30,31
570 DATA 31,29,31,30,31,30,31,31,30,31,30,31
```

月の日数・データ　上は平年，下が閏年

20
メロディーメーカー

X 1には PLAY 文などの音楽機能があり，ゲームなどの効果を出すのに威力を発揮するのですが，データを作るのが大変です。私は音楽の才能が無いため，いつもデータを作るのに四苦八苦していますが，いっそのことピアノやオルガンのように直接演奏して，それがそのままデータになれば良いのに…………。ということで作ったのが，このメロディーメーカーです。これも“AUTO DATA MAKER”と同じように，自己増殖型のプログラムで，データは 60000 行から直接，PLAY 文で保存されます。

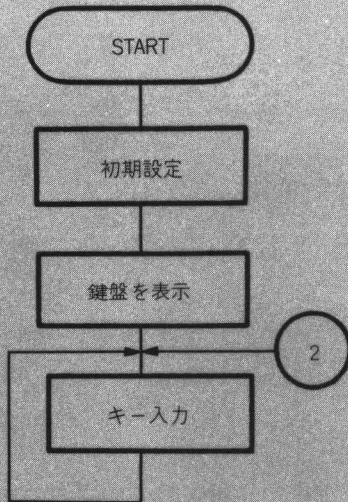
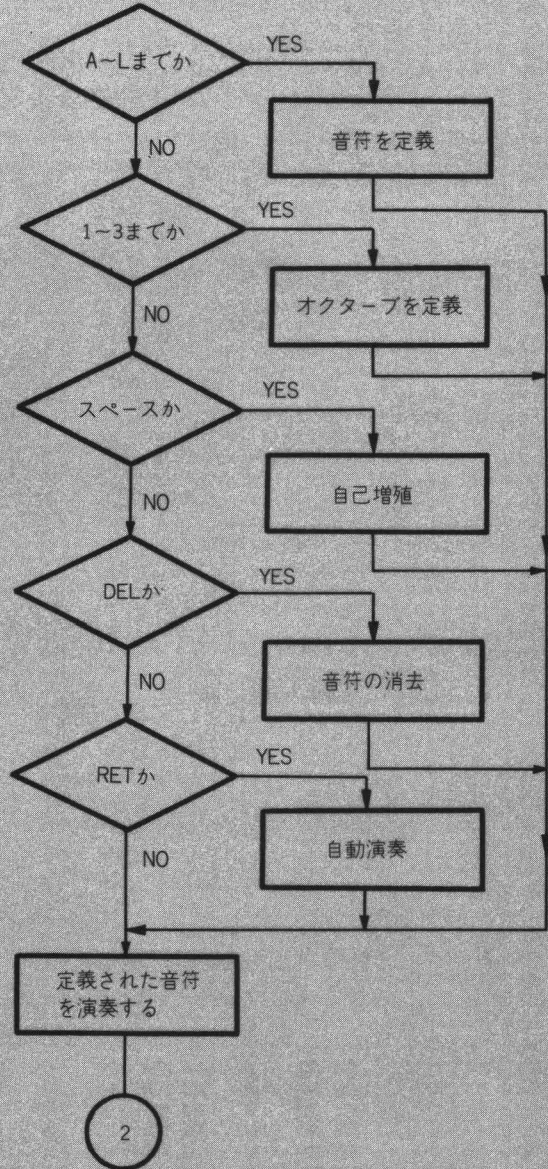
このメロディーメーカーでは，テンポ文などいくつかの機能がありません。みなさんで色々と使いやすくしてみて下さい。

プログラムの使用法

まず，プログラムをRUNしますと，画面いっぱいに鍵盤が表示されます。その下に表示されている英字A～Lまでが，ドレミに対応するキーです。半音は，その斜上というように，ピアノなどと同じような形にしてあります。ファンクションキーの1～3までは，オクターブ指定になっていますが，ここには1～3までの数字が入っているだけですので，テンキーの1～3までも同じようにして使用することができます。休符は0キーです，以上が演奏時に使用するキーですので，演奏してみて下さい。打った音符が上段に表示されたはずです。ここで演奏しそこなった場合は，DELキーを押せば修正が可能です。次に，スペースキーですが，今まで演奏したデータをプログラム化して保存したいときに使用します。最後に RETURN キーですが，これは今までプログラム化したデータを自動演奏するもので，どのように保存されたか確かめるときに使用して下さい。こうして出来上ったデータは 60000 行から入っていますので，必要に応じて修正を加えて完成版にして下さい。60000 行以前をデリートして，ASCII SAVE をしておけば，違うプログラムにつけるときに有効です。

変数表

LN　→データの保存される行番号
I＄　→キー入力用
MU＄→押したキーに対する音符データ
P＄　→今までに演奏したデータ

メロディーメーカー・フローチャート
START
初期設定
鍵盤を表示
2
キー入力
① キー入力
A~Lまでか
YES
NO
音符を定義
1~3までか
YES
NO
オクターブを定義
スペースか
YES
NO
自己増殖
DELか
YES
NO
音符の消去
RETか
YES
NO
自動演奏
定義された音符
を演奏する
2
② 自己増殖
音符データの前に
行番号・PLAYを表示
X1に↑↑CRGOTO90
CRを打たす
END

20. メロディーメーカー

```
1 REM*************************************
2 REM*                                   *
3 REM*  ■■ ﾒﾛﾃﾞｨｰ ﾒｰｶｰ VER 1.0 ■■       *
4 REM*                                   *
5 REM*     [C] COPYRIGHT 1983年 8月       *
6 REM*                                   *
7 REM*  FOR SHARP-X1 BY HEART SOFT       *
8 REM*                                   *
9 REM*************************************
10 KEY1,"1":KEY2,"2":KEY3,"3"              ' 初期設定
20 CONSOLE 0,25:WIDTH 40
30 COLOR 7:CLS:LN=60000!
40 PLAY"V1004":TEMPO 500
50 PRINT"■■ SHARP-X1 ﾒﾛﾃﾞｨｰ ﾒｰｶｰ VER 1.0 ■■";   ' 画面表示
60 LOCATE 0,23
70 PRINT"■■ COPYRIGHT 1983.7. BY HEART SOFT ■■";
80 LOCATE 0,5:GOSUB 430
90 LINE(0,2)-(39,4)," ",BF
100 REM
110 I$=INKEY$                            ' キー入力 (110～310)
120 IF I$="A" THEN MU$="-B"             ' 音符 (120～250)
130 IF I$="S" THEN MU$="C"
140 IF I$="D" THEN MU$="D"
150 IF I$="F" THEN MU$="E"
160 IF I$="G" THEN MU$="F"
170 IF I$="H" THEN MU$="G"
180 IF I$="J" THEN MU$="A"
190 IF I$="K" THEN MU$="B"
200 IF I$="L" THEN MU$="+C"
210 IF I$="E" THEN MU$="#C"
220 IF I$="R" THEN MU$="#D"
230 IF I$="Y" THEN MU$="#F"
240 IF I$="U" THEN MU$="#G"
250 IF I$="I" THEN MU$="#A"
260 IF I$="1" THEN MU$="O3"             ' オクターブ (260～280)
270 IF I$="2" THEN MU$="O4"
280 IF I$="3" THEN MU$="O5"
290 IF I$="0" THEN MU$="R"   ── 休符
300 IF I$=" " THEN LN=LN+10:GOTO 350 ── 自己増殖へ
305 IF I$=CHR$(8) THEN MU$="":P$="":LINE(0,2)-(39,4)," ",BF ── 消去
310 IF I$=CHR$(13) THEN 400 ── 自動演奏
320 P$=P$+MU$:PLAY MU$:MU$=""
330 LOCATE 12,2:PRINT P$
340 GOTO 100
350 REM
360 P$="":MU$=""
370 LOCATE 0,2:PRINT LN;"PLAY";CHR$(34)
380 KEY0,CHR$(30,30,13)+"G.90"+CHR$(13) ── X1に[↑][↑][CR] GOTO 90 [CR]をやらせる
390 END
400 REM
410 P$="":MU$=""
420 GOSUB 60000:GOTO 100
430 REM
440 PRINT"  F1=ｵｸﾀｰﾌﾞ3  F2=ｵｸﾀｰﾌﾞ4  F3=ｵｸﾀｰﾌﾞ5    ";   ' 鍵盤のデータ (440～600)
450 PRINT"                                       ";
460 PRINT"        (#ﾄ) (#ﾚ)   (#ﾌ) (#ｿ) (#ﾗ)       ";
470 PRINT"          E    R      Y    U    I        ";
480 PRINT"  ┌┬──┬──┬─┬─┬──┬──┬─┬─┬─┬─┬──┬──┬┐  ";
490 PRINT"  ||■■ ■■| |■| |■■ ■■| |■| |■| |■■ ■■||  ";
500 PRINT"  ||■■ ■■| |■| |■■ ■■| |■| |■| |■■ ■■||  ";
510 PRINT"  ||■■ ■■| |■| |■■ ■■| |■| |■| |■■ ■■||  ";
520 PRINT"  ||■■ ■■| |■| |■■ ■■| |■| |■| |■■ ■■||  ";
530 PRINT"  ├■■ ■■└─┘■└─┘■■ ■■└─┘■└─┘■└─┘■■ ■■┤  ";
540 PRINT"  |■■■ ■■■ ■■■ ■■■ ■■■ ■■■ ■■■ ■■■ ■■■|  ";
550 PRINT"  |■■■ ■■■ ■■■ ■■■ ■■■ ■■■ ■■■ ■■■ ■■■|  ";
560 PRINT"  |■■■ ■■■ ■■■ ■■■ ■■■ ■■■ ■■■ ■■■ ■■■|  ";
570 PRINT"  |■■■ ■■■ ■■■ ■■■ ■■■ ■■■ ■■■ ■■■ ■■■|  ";
580 PRINT"  └──────────────────────────────────┘  ";
590 PRINT"   A    S    D    F    G    H    J    K    L    ";
600 PRINT"  (-ｼ) (ﾄ) (ﾚ) (ﾐ) (ﾌ) (ｿ) (ﾗ) (ｼ) (+ﾄ)  ";
610 RETURN
60000 REM
65530 RETURN
```

後書き

一時期，一部の人達のものであったマイコン，パソコンが最近では広い層の多くの人達が使うようになったのは，大変喜ばしいことであります。近頃は，ゲーム，シュミレーション，ビジネス用などの多くのパッケージソフトウェアが出て，広範囲な使われ方が出来るようになりました。しかし，反面，市販のパッケージソフトは汎用性を持たせているために，どうしても変更したい箇所が出て，〝BASIC言語を習得するには，どのような方法が良いか？〟などの声が多いのも事実であります。

BASIC 言語を覚える一番の近道は，前書きにも書いたように，〝習うより慣れろ〟で，本書のゲームプログラムを入力，実行しプログラムを変更していくことにより，オリジナルなプログラムに作り変えて行くのが能率の良い，習得方法になると思います。また，ゲームソフトは豊富に揃っているものの高価であるために，なかなか入手出来ないのが現状です。本書では，約20のゲームプログラムを集録しており，楽しむことが出来ると思います。是非，本書により，ゲームを楽しみ，かつまた，BASIC 言語の参考書にしていただき，本書が，あなたのマイコンライフに少しでも，役に立てば幸いだと思います。

HART　SOFT
Y. TAKASE

■御　案　内

本書に掲載している20本のゲームプログラムを収録した，カセットテープを読者に5,000円（送料含む）で頒布いたします。

御希望の方は，下記申込書に必要事項を御記入のうえ現金書留にてハート電子産業株式会社まで，御申込みください。

〔対象機種〕　シャープ パソコンテレビＸ１

〔メディア〕　標準カセットテープ

〔価　　格〕　5,000円（送料含む）

〔申 込 先〕　〒221

神奈川県横浜市神奈川区東神奈川２－40－９

クインビル　７Ｆ

ハート電子産業株式会社

TEL.045-441-0190

------------------------------------ 〈キリトリ線〉 ------------------------------------

プログラム申込書

年　　月　　日

パソコンテレビＸ１ゲームプログラム(II)・カセットテープを申込みます。

〔御住所〕

〔お名前〕

〔お電話〕